STUDY AND ANALYSIS OF DMC AND APPLICATION IN D.C. MOTOR SPEED CONTROL

THESIS SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF

MASTER OF TECHNOLOGY
IN
ELECTRONICS AND INSTRUMENTATION ENGINEERING

BY

MOHIT SRIVASTAV (212EC3380)

Under the Guidance of

Prof. TARUN KUMAR DAN

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA

2013-2014

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA

CERTIFICATE

This is to certify that the thesis entitled, "**Study and Application of Dynamic Matrix Control**", submitted by **Mr. Mohit Srivastav** in partial fulfillment of the requirements for the award of Master of Technology Degree in **Electronics And Instrumentation Engineering** at National Institute of Technology - Rourkela is an authentic work carried out by him under my supervision. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university / institute for the award of any Degree or Diploma.

Date

Prof. Tarun Kumar Dan
Dept. of Electronics and Communication Engineering
National Institute of Technology
Rourkela-769008

ACKNOWLEDGEMENT

I take this opportunity to express my deepest gratitude and sincere thanks to my respected supervisor **Prof. Tarun Kumar Dan**, Department of Electronics and Communication Engineering, without his guidance, I could have never completed my project in this stipulated period of time. His invaluable support, guidance, motivation and encouragement throughout the period has resulted this accomplishment. I would also like to thank all the Professors and members of the Department of Electronics and Communication Engineering for their generous help in various ways for the completion of the thesis. I also extend my thanks to the fellow students for their friendly co-operation.

Mohit Srivastav (212EC3380)

ABSTRACT

The Dynamic Matrix Control (DMC) is a subset of Model Predictive Control (MPC) algorithm introduced in early 1980s. It is a proven method which gives good performance. It is also the only technique which is able to consider model constraints.

This project deals with the study, design and application of Dynamic Matrix Control for an unconstrained single input and single output system. The step response of the plant has been used for model prediction. The objective function used is "Least square objective function" which is based on squares of difference between predicted output and actual plant input .The strategy applied in DMC is to optimize the objective function, for this we use a unique expression of feedback matrix k (which is obtained through simple calculus).

Further the response of plant being controlled by DMC controller is obtained which are significantly better than the step response of the plant. Then study of the effect of DMC parameters P (prediction horizon), N (model length), M (control horizon length), W1(error weight matrix), W2(control weight matrix) on plant response have been done. After this, effect of disturbance on the plant response and effect of weighting tuning parameters is also discussed. A plant with random noise and it's control is also discussed.

The last chapter of this thesis holds the application domain, which shows the applicability of DMC through a simple DC motor. First, modelling of DC motor is done, then it is further tested in SIMULINK. Then the same is used for getting appropriate set-point trajectory of motor output at a set armature voltage for no load case. Further in this section, control of motor with no load and with load, both the cases are discussed. Further, capability of DMC with different load values is tested. Next comes testing DMC control law against varying load and then DMC is tested against variation of loading time. In all the above given scenarios DMC performs very well. All the above analysis is supported by MATLAB simulations.

List of Figures

INDEX

Chapter 1

Introduction

1. INTRODUCTION

1.1 What is DMC:

This algorithm was developed firstly by Shell Oil engineers in late 1970's and was intended for its use in petroleum refineries. Dynamic Matrix Control or in short DMC is a control algorithm designed specially to predict the future response of a plant [1]. Now-a-days its applications are found in a wide variety of areas including aerospace applications, chemicals, food processing and automotive.

1.2 MOTIVATION:

Mostly controllers (which are other than model predictive) are designed in analog domain and then they are executed by the microcontrollers or microprocessors. This all needs conversion of controller transfer function from continuous to digital domain. They may also need discretized model of plant. In this conversion some approximations are used, so we can conclude that the use of conventional controllers requires an additional step of conversion for implementation.

But DMC is inherently discrete time control technique. It do not require any Conversion of plant transfer function and even do not require any plant model. It simply uses the sampled data taken from the step response of the plant before starting the control action. This data acts as a model for plant.

1.3 How does DMC works:

The main motive of Model Predictive Control is to find the input signal that best satisfies to a given performance criterion, it predicts how the system will behave if the signal is applied.
This algorithm uses the samples of step response of plant to capture plant's nature and solves a control problem for an optimal control action to track the given input. Following approximation is used for plant response [1]:

$$Y(k) = \sum_{i=1}^{\infty} S_i \, \Delta u_{k-i}$$
$$= S_1 \Delta u_{k-1} + S_2 \Delta u_{k-2} \ldots\ldots + S_N \Delta u_{k-N}$$

It's work can be summarized into following steps [5]:

1 Find step response coefficients matrix

S = [s1,s2,s3,s4…….sn].

2 Find Sf and Sp matrix from S.

3 Declare W1 and W2 matrix and calculate feedback gain

Matrix:

$K = (S_f{}^T \, W1^T \, W1 \, S_f + W2^T \, W2)^{-1} S_f{}^T \, W1^T \, W1$

4 find error E = r – y_free.

5 Find required change in control weight at instant k:

$\Delta u_f(k)$ = K*E and update the value of controller output as:

u(k+1) = uinitial + $\Delta u_f(k)$.

6 Get the response of plant at instant k+1:

y(k+1) with respect to u(k+1) and set point value r.

7 Get a prediction of future response of plant from approximation is used for plant response as:

Ymod(k+1) = S(1)* $\Delta u_p(k)$ + Sp* $\Delta u_p' + S(N) * \mathrm{u}(\mathrm{k} - \mathrm{n} + 1)$

8 Now, update this value as new value of y_free.

Now, repeat step 4 to step 7 for each new value of time instant.

Chapter 2

Model Predictive Control (MPC)

2.0 Model Predictive Control (MPC)

2.2 Primary components of MPC

Main components of Model Predictive Control method can be named as [5]:

1. The Process Model
2. The Cost Function
3. The Optimizer

The *process model* includes the step response coefficients of the controlled process and this process model is used to predict the response of the process with respect to various manipulated control variable values. Then minimization of *cost function* ensures that the error is reduced and other parameters such as control moves' magnitude is penalized as per constraints on plant. In the last step through different *optimization* techniques application results in an output and this output gives the input sequence for the next prediction horizon.

Aim of *Model Predictive Control* is to predict the input signal to the plant that can at best satisfy to some criterion. At each time step, k, an optimization problem is solved. An objective function (usually quadratic) is minimized, over a prediction horizon of P time steps by a selection of manipulated variable moves over a *control horizon* of *M* control moves.

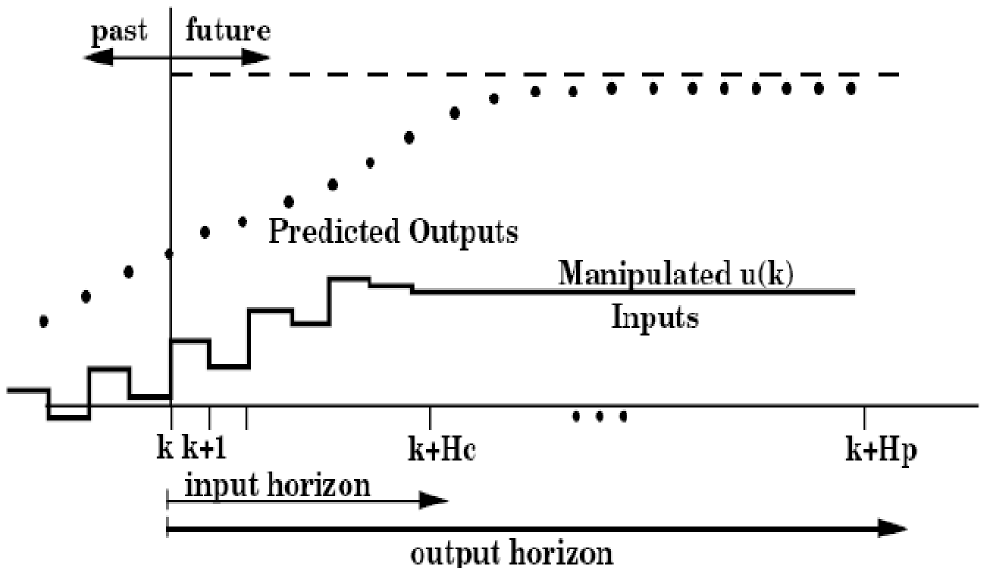

Fig.1. Receding Horizon Control

Only the first move is applied on the plant, although *M* moves are optimized. After u_k is implemented, the measurement at the next time step, y_{k+1} is obtained, A correction for model error is performed, since the measured output y_{k+1} will, in general, not be equal to the model predicted value. Then a new optimization problem is then solved, again, over a prediction horizon of *P* steps by adjusting *M* control moves. This approach is also known as "*Receding Horizon Control*".

DMC is based on the step response model, which has the form [1] :

$$\widehat{y_k} = \sum_{i=1}^{N-1} S_i\, \Delta u_{k-i} + S_N \Delta u_{k-N} \quad (1)$$

$$= S_1 \Delta u_{k-1} + S_2 \Delta u_{k-2} \ldots\ldots + S_{N-1} \Delta u_{k-N+1} + S_N \Delta u_{k-N}$$

Where $\widehat{y_k}$ is the model prediction at time step k, and Δu_{k-N} is the manipulated input N steps in the past. Since, model-predicted output is unlikely to be same as actual measured output of plant y_k, here we get another term "additive disturbance" which causes this inequality. It can be termed as d_k. So, "corrected prediction" can be written as

$$\widehat{y_k}^{c} = \widehat{y_k} + d_k.$$

For j^{th} step in future $\widehat{y_{k+j}}^{c} = \widehat{y_{k+1}} + d_{k+1}.$

$$\widehat{y_{k+j}}^{c} = \sum_{i=1}^{j} S_i\, \Delta u_{k-i+j} + \sum_{i=j+1}^{N-1} S_i\, \Delta u_{k-i+j} + S_N u_{k-N+j} + d_{k+1}. \quad (2)$$

Note that here two assumptions are taken in consideration, they are as:

1. Constant additive disturbance assumption :
   $\widehat{d_{k+j}} = \widehat{d_{k+j-1}} = \ldots\ldots = d_{k+1}.$
2. There are no control moves beyond control horizon of M steps :
   $\Delta u_{k+M} = \Delta u_{k+M+1} = \ldots\ldots = \Delta u_{k+p-1} = 0.$

Now, if we try to write (2) in *matrix form* for prediction horizon of P and control moves of M steps we get [1]:

$$\underbrace{\hat{Y}^{c}}_{\substack{corrected \\ predicted\ outputs}} = \underbrace{S_f \Delta u_f}_{\substack{effect\ of \\ current\ and\ future \\ moves}} + \underbrace{S_{past} \Delta u_{past}}_{effect\ of\ past\ moves} + \underbrace{\hat{\mathrm{d}}}_{\text{predicted disturbances}} \quad (3)$$

$$\underbrace{r - \hat{Y}^{c}}_{\substack{corrected \\ predicted\ error\ E^{c}}} = \underbrace{r - \{\ S_{past} \Delta u_{past} + \underbrace{\hat{d}}\ \}}_{\substack{unforced\ error(if\ no\ current\ and \\ and\ future\ control\ moves\ were\ made),\ \ E}} - \underbrace{\mathrm{S_f \Delta u_f}}_{\substack{\text{effect of} \\ \text{current and future} \\ \text{moves}}} \quad (4)$$

Now, we take a least-square objective function [1]:

$$\emptyset = \sum_{i=1}^{P} (\ e_{k+1}^{c})^2 + \ w \sum_{i=0}^{M-1} (\ \Delta u_{k+1}\ )^2 \quad (5)$$

Where, both first and second summation can be expressed in *matrix form* as:

$$\sum_{i=1}^{P} (\ e_{k+1}^{c})^2 = (\ E^{c})^{T}\ E^{c} \quad (6)$$

$$w \sum_{i=0}^{M-1} (\ \Delta u_{k+1}\ )^2 = (\Delta u_f\ )^{T}\ W \Delta u_f \quad (7)$$

Therefore, from (4),(6),(7) we can express (5) in *matrix form* as:

$$\emptyset = (\mathrm{E} - S_f \Delta u_f)^{T} (\mathrm{E} - S_f \Delta u_f) + (\Delta u_f\ )^{T}\ W \Delta u_f \quad (8)$$

Solution for the minimization of the above given optimization function is [1]:

$$\Delta u_f = \underbrace{(\ {S_f}^{T} S_f + \mathrm{W}\ )^{-1}\ (S_f\ )^{T}}_{K}\ \mathrm{E} \quad (9)$$

Since only current control move is applied (or for M=1), we use first row of K matrix

$$\Delta u_f = K_1 \text{ E} \tag{10}$$

2.2 Basic Structure of MPC :

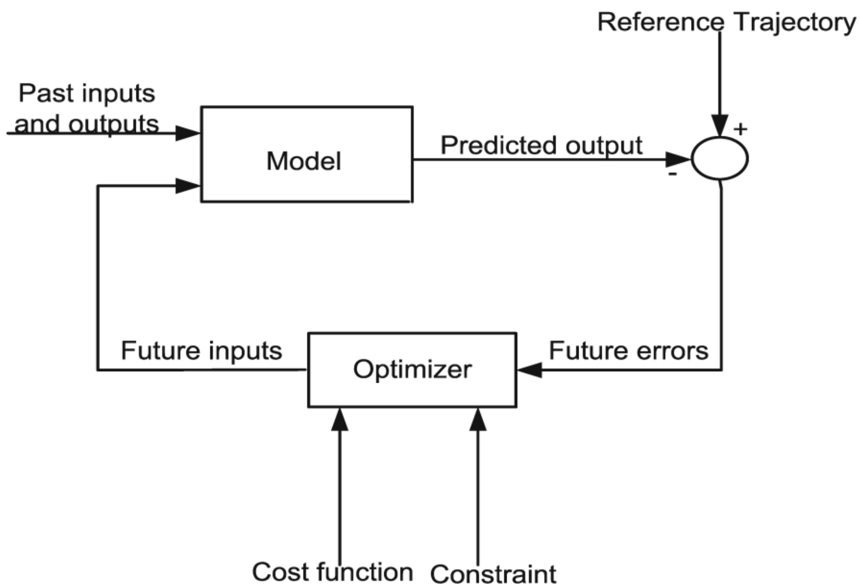

Figure 2: Basic structure of MPC [4]

2.3 Introduction to DMC tuning parameters:

Tuning parameters of a dynamic matrix controller can be given as [5]:

N Model length

P Prediction horizon

M Control horizon

W1 Error weight matrix

W2 Control weight matrix

Model length (N):

This parameter defines the number of samples of step response of the plant to be taken by the controller.

Prediction horizon (P):

This parameter defines the number of future response predictions of the plant to be done by controller for every new value of controller output.

Control horizon (M):

This parameter defines the number of future control actions calculated by the controller to get the desired response.

Error weight matrix (W1):
This matrix holds the value for error term in control law. It is an important parameter for DMC tuning.

Control weight matrix (W2):
This matrix holds the value of weight assigned to control moves.

2.4 DMC tuning steps:

- The model horizon N should be selected such that N*Δt $\geq$ open loop settling time. Δt is the time interval between successive intervals. Normally, the value of N is taken from 20 – 70. Now, if t_0 is the dominant time constant of system, then the settling time would be around $5t_0$. If we take around 50 steps then $5t_0$ is approximately equal to 50Δt.
- P should cover the main dynamic part of step response. If P is increased results in more gradual control action but increases computational time.
- For monotonous dynamic characteristics M = 1 to 2.
- For oscillation dynamic characteristics M = 4 to 8.
- Generally, W1= I (identity matrix) and W2= p I , where p is a constant . Larger values of p penalize the control moves making the system response sluggish.
- If P>>M then weight p $\approx$ 0, this is because if p has integral values then control moves become very large and aggressive.

2.5 Advantages of MPC

There are various reasons for MPC to be so much successful. Some of the major contributing points are:

1. It can the handle the structural changes in the plant or process.

2. It allows consideration of plant constraints therefore profit is more.

3. Actuator limitations can also be considered with the plant constraints.

4. Since, implemented on microcontrollers which are very fast in calculations, complex control optimization algorithms can be implemented.

5. Non-minimal phase, system with inverse response and unstable processes can be controlled easily.

6. It can handle MIMO systems and multivariable control problems naturally.

2.6 Applications of MPC

- Servo mechanism
- Pulp and paper plant
- Aviation
- Aerospace
- Distillation column
- Robotic arm … etc.

Chapter 3

Cost Function

3. Cost Function

3.1 Cost Function or Objective Function

Cost function gives a measure of performance of the controller. The objective function is the performance criterion for controller. It is required, that the controller makes plant to follow a particular set point trajectory with satisfying some performance criterion. This is achieved by minimizing the cost function or objective function. Every objective function contains the respective sub functions which are required to be penalized or preferred.

3.2 Types of Cost Function

There are various different kinds of objectives functions. Few of them are Standard Least-Squares or Quadratic Objective Function, Absolute Value Objective Function, etc.

Quadratic Objective Function:

This objective function is summation of squares of the predicted errors and the control moves. Here, Predicted errors are the differences between model predicted outputs and the set points. Control moves are the applied or predicted changes in control action by the controller. Say, a quadratic objective function has a prediction horizon of length 2 and a control horizon of length 1 then it's expression can be written [1]:

$$Ø = (r_{k+1} - y_{k+1})^2 + (r_{k+2} - y_{k+2})^2 + \mathrm{w}\,\Delta u_k^2$$

Here, y represents the *model predicted output*, r is the *set point*, Δu is the *change in manipulated input* from one sample to the next and w is a *weight* for the changes in the manipulated input. Sample time for which the objective function is being taken is indicated by the subscripts where, k denotes the *current sample time*. For a prediction horizon of P and a control horizon of M, the least Squares objective function can written as [1]:

$$Ø = \sum_{i=1}^{P}(r_{k+1} - y_{k+1})^2 + \mathrm{w}\sum_{i=0}^{M-1}\Delta u_{k+1}^2$$

Absolute Value Objective Function:

This is another possible objective function which simply takes a sum of the absolute values of the predicted errors and control moves. Though this type of objective function is fairly simple as compared to quadratic objective function, the latter is more suitable in its approach handling the non -linear process models. For a prediction horizon of 2 and a control horizon of 1, the absolute value objective
function is [1]:

$$Ø = |(r_{k+1} - y_{k+1})| + |(r_{k+2} - y_{k+2})| + \mathrm{w}\,|\Delta u_k|$$

Quite similar to quadratic objective function, it has the following general form for a prediction horizon of P and a control horizon of M [1]:

$Ø = \sum_{i=1}^{P} |(r_{k+1} - y_{k+1}) | \quad + \text{w} \sum_{i=0}^{M-1} | \Delta u_k|$

The optimization problem in hand is solved as a result of minimization of the objective function. This is obtained by adjusting the M control moves, subject to modeling equations and constraints on the inputs and outputs.

Hence, $\quad Min_{\Delta u_k \ldots\ldots, \Delta u_{k+M-1}}\ Ø$

3.3 Cost function used

In the present project we are using quadratic objective function. It mainly contains two parts one which defines error (deviation of plant response from set point trajectory) and other defines size of control moves use to control the system (or controller output). With both of the parts a weight is present which is used for effecting the consideration of error and amplitude of control weights. W1 defined in tuning parameters is associated with error part, it is known as *Error Weight* and W2 is used for penalizing the control moves, it is known as *Control Weight.*

Chapter 4
MATLAB implementation of DMC

4. MATLAB implementation of DMC

4.1 Response of a plant being controlled by DMC :

Consider a *plant transfer function* given below:

$$G(z) = \frac{0.2713\ z^{-3}}{1 - 0.8351\ z^{-1}}$$

Controller parameter values:

Prediction horizon length P=10,

Control horizon length M=1,

Model length N=50,

Control weight W2=0,

Error weight W1=1.

MATLAB parameters :

Delt = 0.1 sec

Timesp = 1 sec

Tfinal = 5 sec

Results :

Step response coefficients of plant :

In the above program the value of N=50. This shows that the number of samples of the plant response, to be taken by the controller. Here in the fig.3 the step response coefficients of the plant are shown. They sampled at the sampling time period of 0.1 sec.

Response of plant being controlled by DMC and DMC control output:

Here , in the below given fig.4 the response of the plant is shown .We can see that it's rise time is much better than the actual step response without the controller. The respective controller output is also shown which has very gradual control actions and prevent it to be prone to modeling uncertainty.

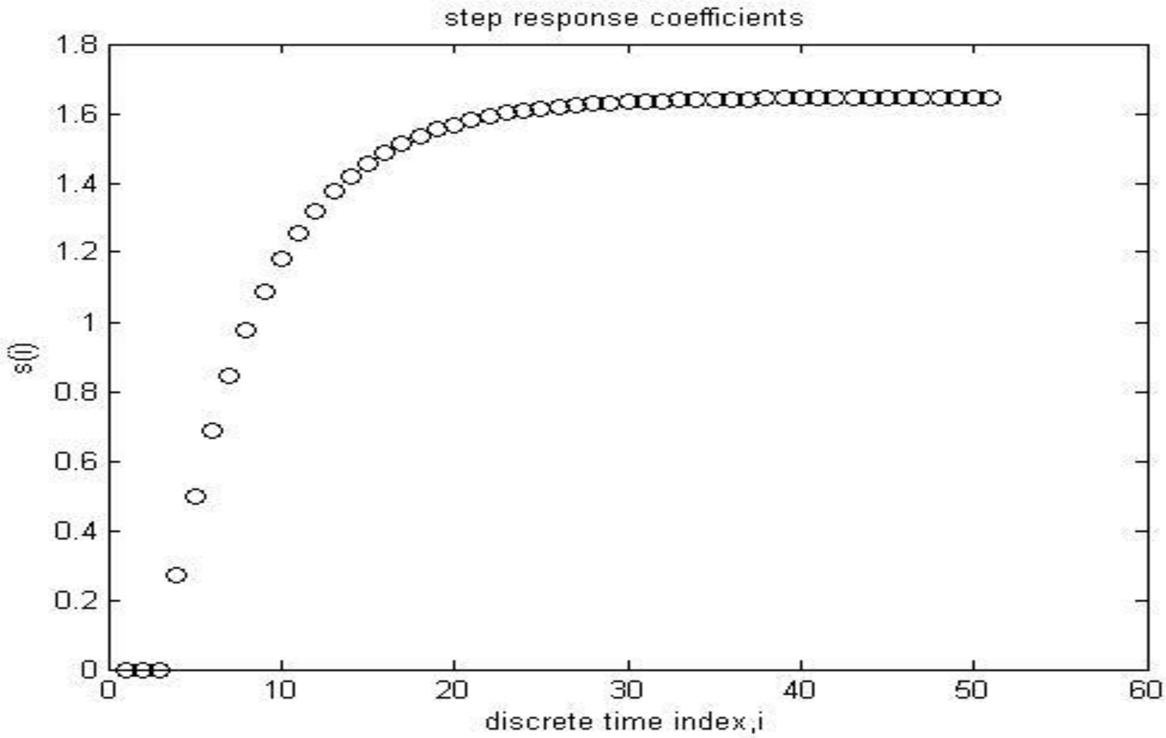

Fig3. step response coefficients of plant

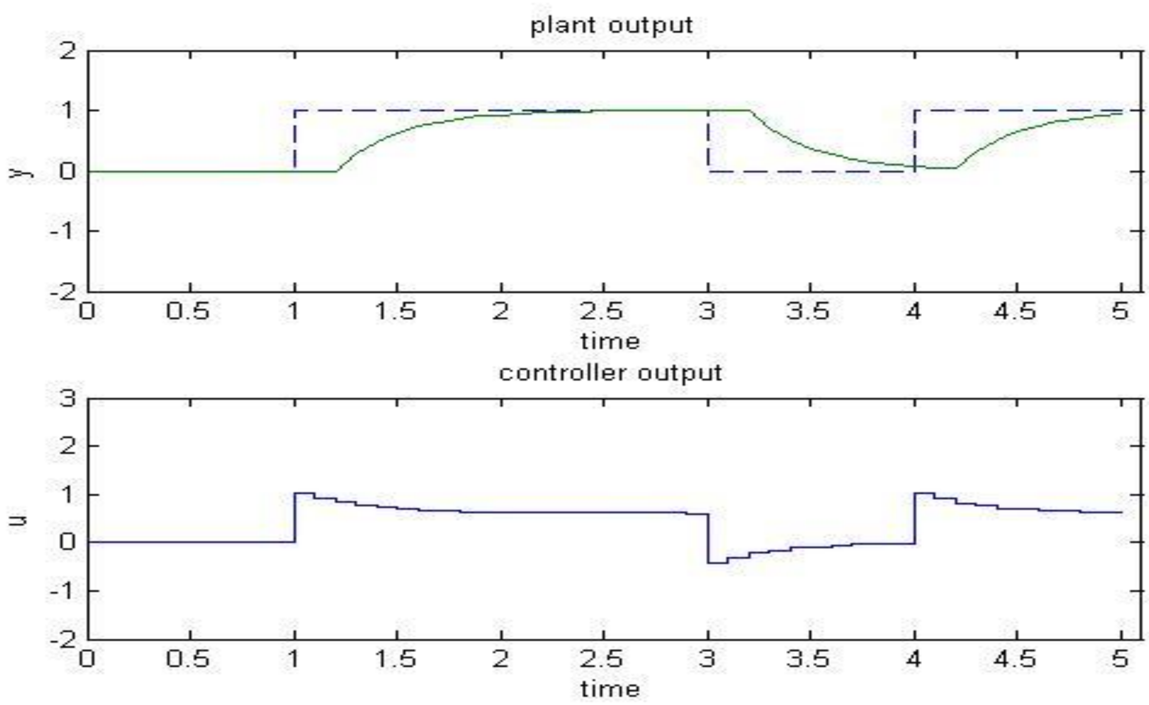

Fig 4 . Plant output and DMC controller output

4.2 Effect of DMC tuning parameters on plant response:

Consider the same plant and same control parameters as given in above program.

Effect of Prediction Horizon (P):

Prediction horizon defines the predicted future output values. In the below given fig. 5 the value of P is varied from 10 to 3 and other values are kept constant M=1, N=50, W2=0, W1=1.

As we can see in the response that when value of P is increased the output reaches the set point is very slowly and for lower values of P response is faster. For lower values of P, higher control action is required and it is more prone to modeling uncertainty.

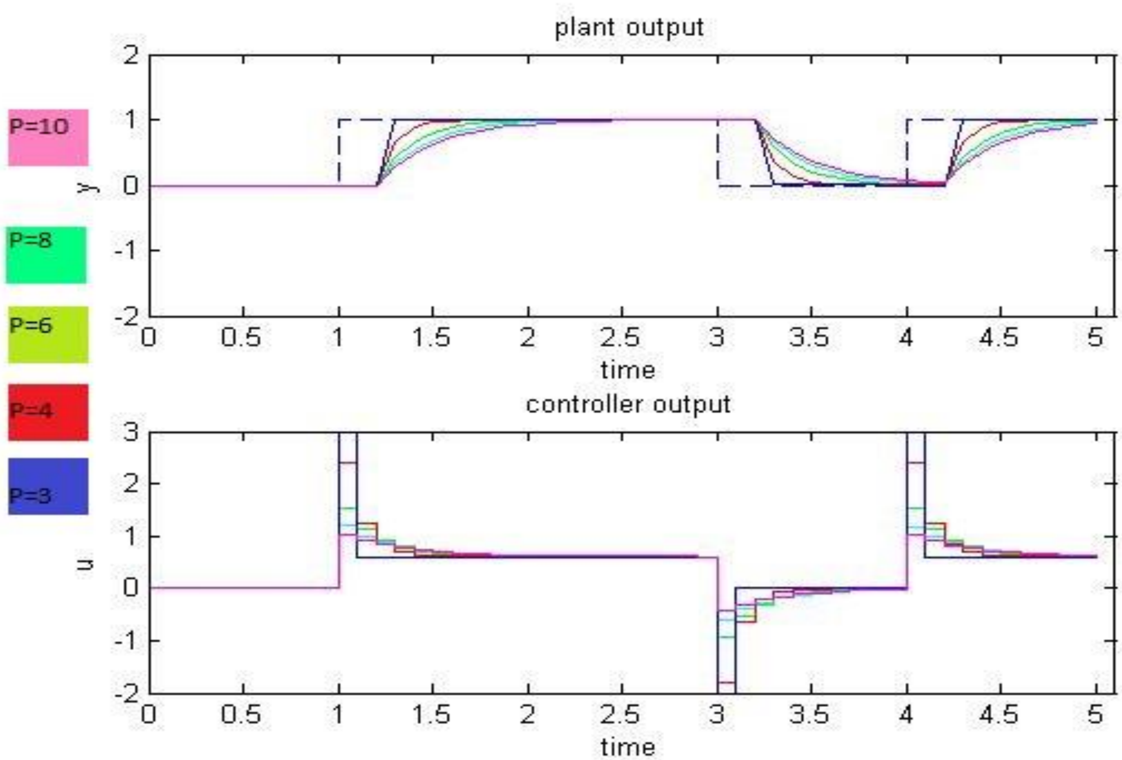

Fig..5 Effect of prediction horizon P on plant output and controller output

Effect of control horizon (M):

Control horizon defines the number of future control actions calculated by controller. In the below given fig. 6 the values of M = 1, 3, 5 in different responses and other values are kept constant P=10, N=50, W2=0, W1=1.As we can see in the response for smaller values of M, plant response reaches to set point value very slowly and controller output in very gradual and for higher values of M , the response becomes faster in reaching the set point value.

But, for higher values of M, controller becomes more sensitive to external noises. So, higher values of M are not chosen usually.

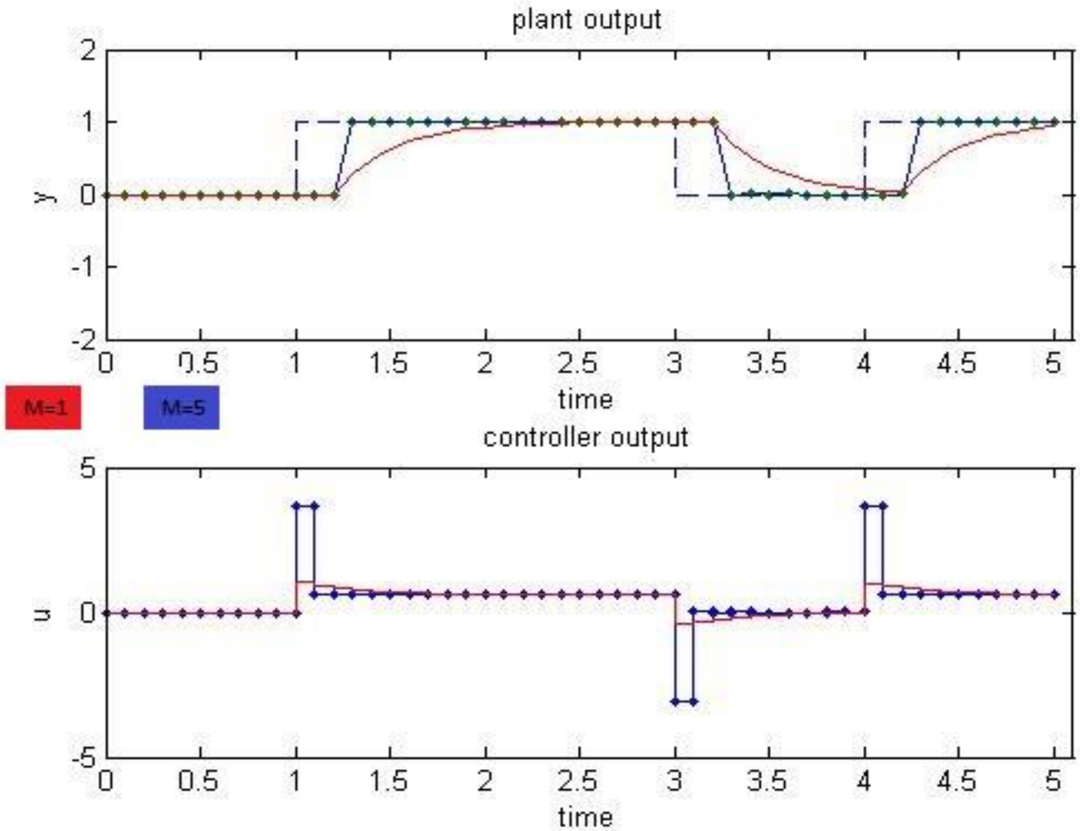

Fig 6. Effect of control horizon length M on controller response and plant response

Effect of model length (N):

Model length defines the number of plant's step response samples to be taken by controller. Model length must be such that full dynamic characteristic of plant is captured. If it is not adequate then controller has model errors and poor performance. In the below given fig.7 we take N=30, N=50, N=10. We can see

that for N=50 and N=30 the response of plant is almost same and has no sign of instability. But for N=10 we can see that response becomes instable which is due to small value of N which is incapable to hold sufficient dynamics of plant response.

Effect of error weight matrix W1 and control weight matrix W2 :

W1=diag[w1 w1 w1…P-times]

and

W2=diag[w2 w2 w2 …M-times]

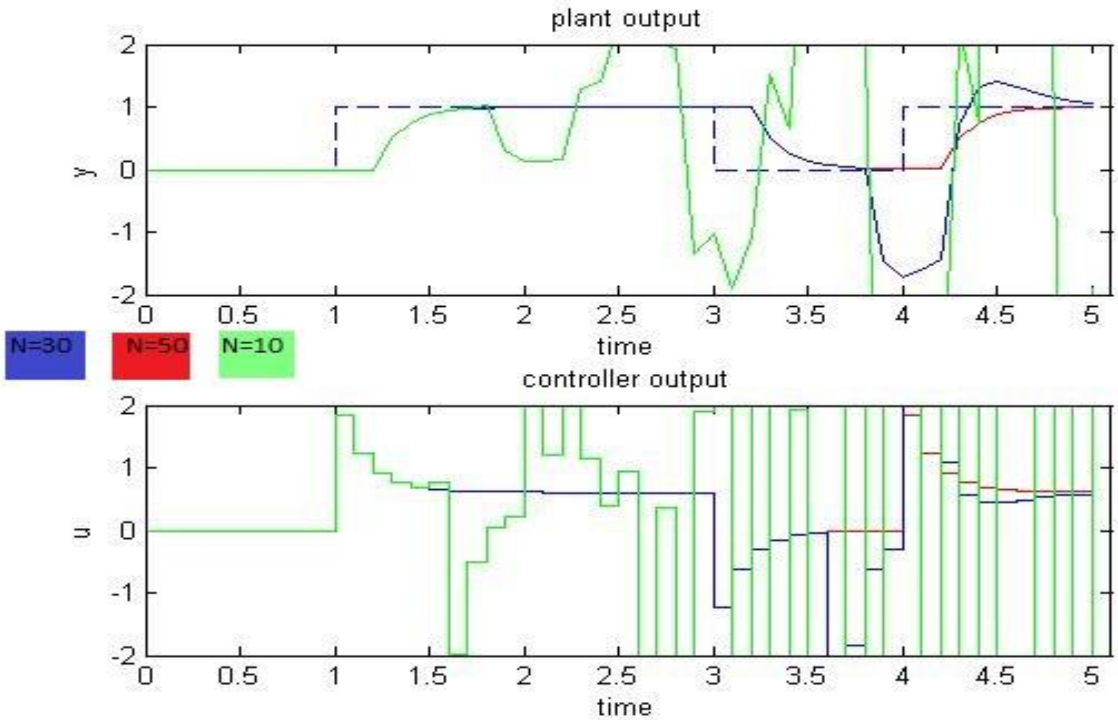

Fig 7. Effect of model length (N) on plant response and controller output

w1 and w2 are very important tuning parameters.w1(ERROR WEIGHT) is used mainly for controlling the rise time of response.w2 (CONTROL WEIGHT) is used mainly for getting gradual control signal.

Here we take a system transfer function:

$$G(z) = \frac{0.2713}{z^3 - 0.8351\ z^2}$$

And disturbance = $\frac{0.1}{z-0.05}$

The system response and disturbance are superimposed and therefore the samples taken by controller become erroneous. Fig. 8 shows the step response coefficients of plant with disturbance. Here, we see that the obtained response is a little higher in magnitude than response shown in fig. 3.

Now when the step response coefficient values shown in fig. 8 are fed to DMC controller having all parameter values set to: P=10, M=1, N=50, W2=0 and varying W2 as: W1=1,4,7.We get following responses of plant and controller:

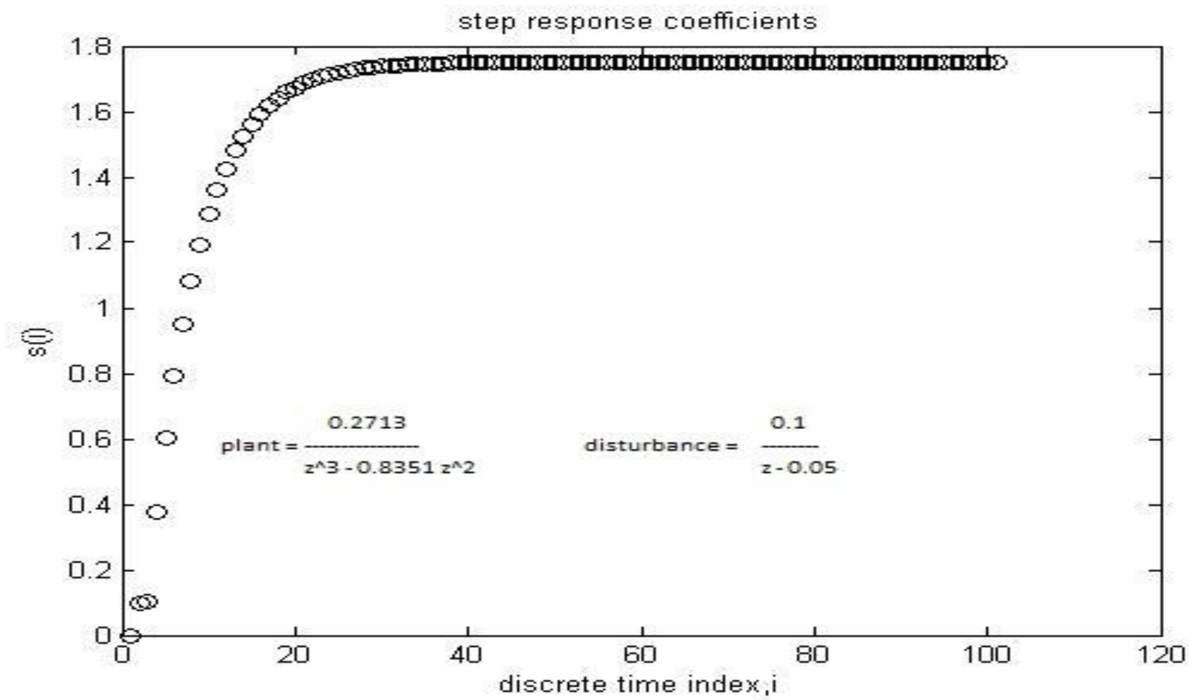

Fig.8 step response coefficients of plant with disturbance

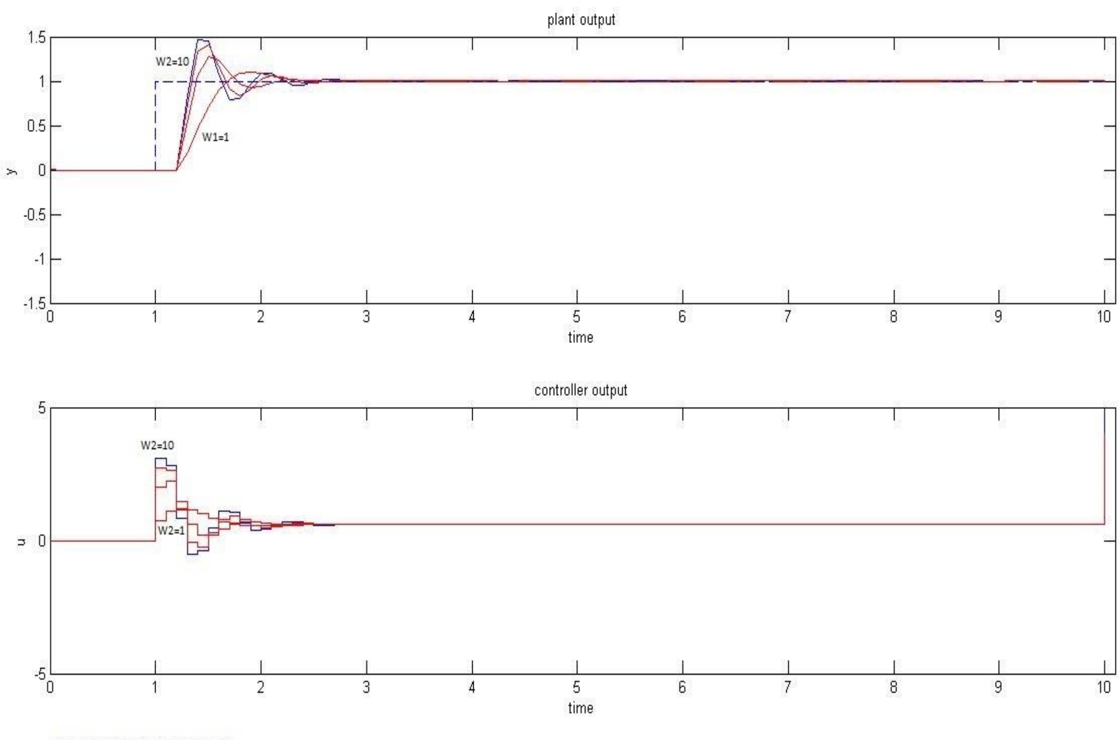

Fig .9 Plant response and controller output for w1=1,4,7, P=10, M=1, N=50, W2=0

We can see in fig. 9 that if the value of W1 is increased it results in faster response but that results in overshoots and also has large values of control moves. Whereas, we can see the system response in fig 10. if we vary W2 as : w2= 1,4,7 and keep all the other parameters constant :

M Model length = 50

P Prediction horizon = 10

M Control horizon = 1

W1 Error weight matrix = 1

W2 Control weight matrix = varying

We can see in fig 10 that the plant response becomes faster for lower values of W2 but it has much impact on control moves. So, we can conclude that this parameter must be use for producing gradual response of control moves.

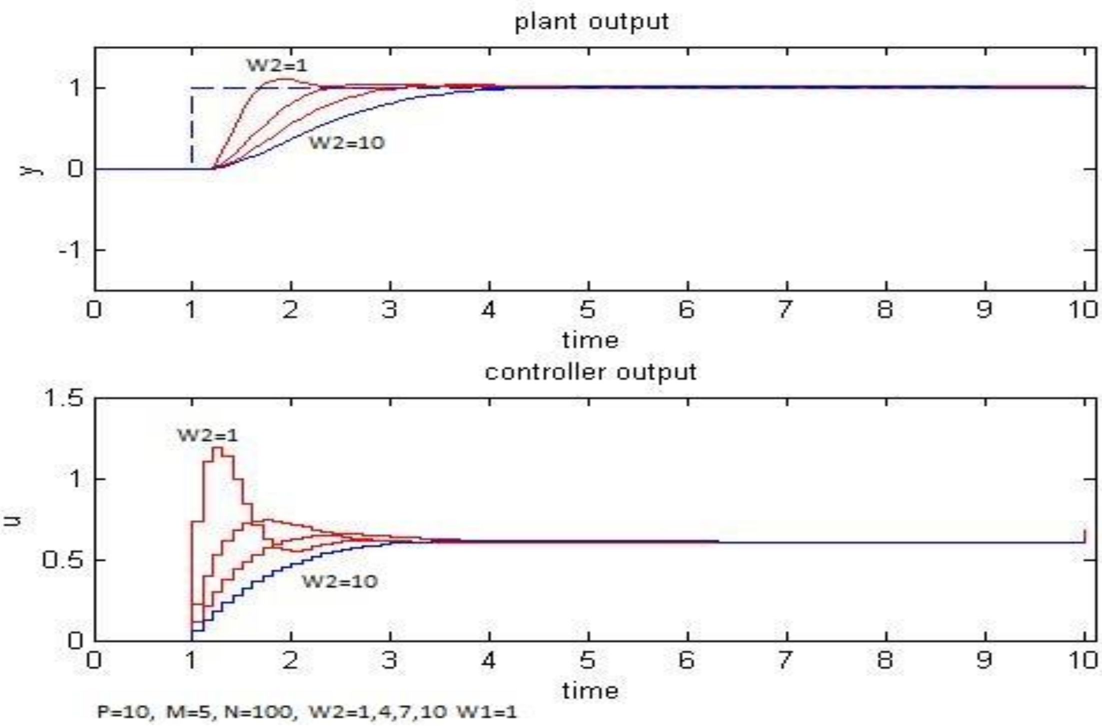

Fig.10 plant response and controller output w2=1,4,7

4.3 Finding appropriate value of W2 to get a response with low rise time and gradual control moves:

As we know that for lower values of P we get faster response but at the expense of much more control action. But what if we cannot afford high magnitude control input as in the case of constrained systems.

Error weight value can help us in getting desired response with gradual control input. We can get a similar response with gradual control action with an appropriate value of W2.

We can set fractional values of w2 to get similar response with lesser control action. Negligible overshoots may be present in the response by this method. In the below given simulation fig. 11 we can see that for w2=0.1 control response is very similar to the response for w2=0. As we increase w2 the overshoots increase.

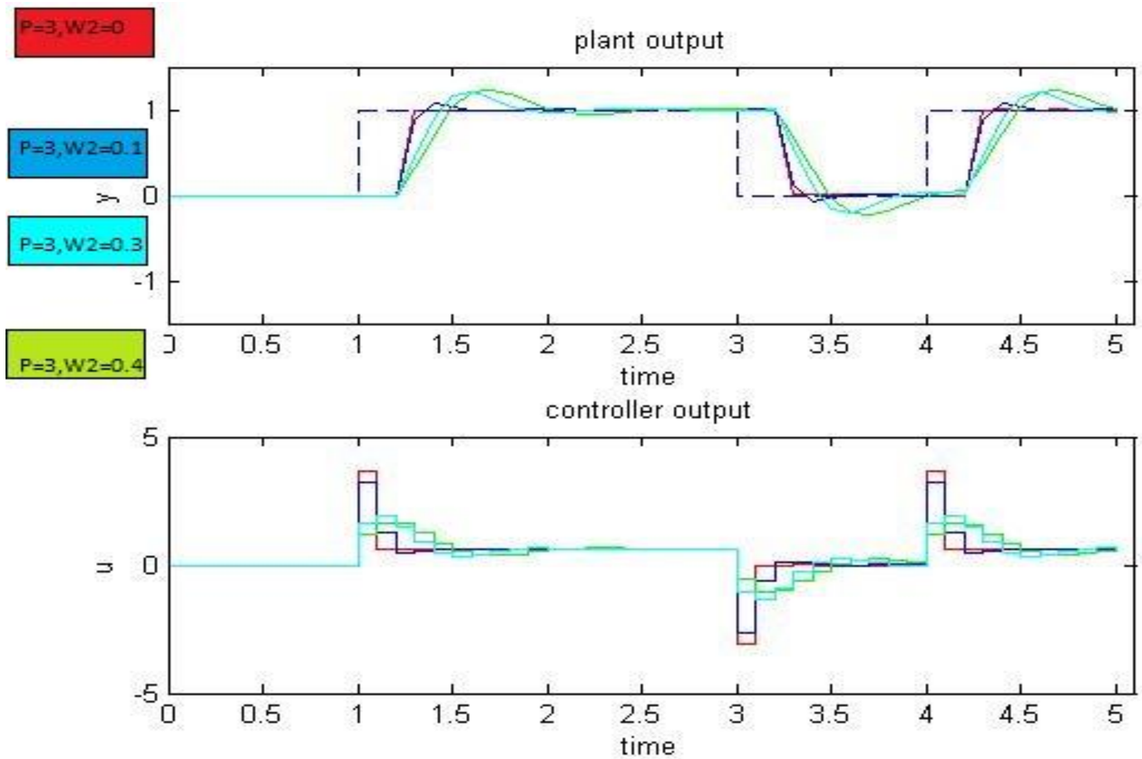

Fig 11 Obtaining a faster response with gradual control moves

Chapter 5
System-noise and Disturbances

5. System Noise and Disturbances

5.1 Effect of noise on controller performance with different values of control horizon length:

For analyzing a plant with noise consider the following:

$G(z) = \frac{0.2713}{z^{-3} - 0.8351\ z^{-2}}$ = plant

Disturbance : D = [0.1 -0.1 0.1 -0.1 0 ………… 0]

Controller parameter values:

P=10, M=1,2 , N=50, W2=0, W1=1.

MATLAB parameters :

Delt = 0.1 sec

Timesp = 1 sec

Tfinal = 5 sec

Here, the fig. 12 shows the response of system for M=1 and has no disturbance added. But in fig. 13 system has added noise and we can see that it's response shows irregular behavior .We can see that for system having disturbances with

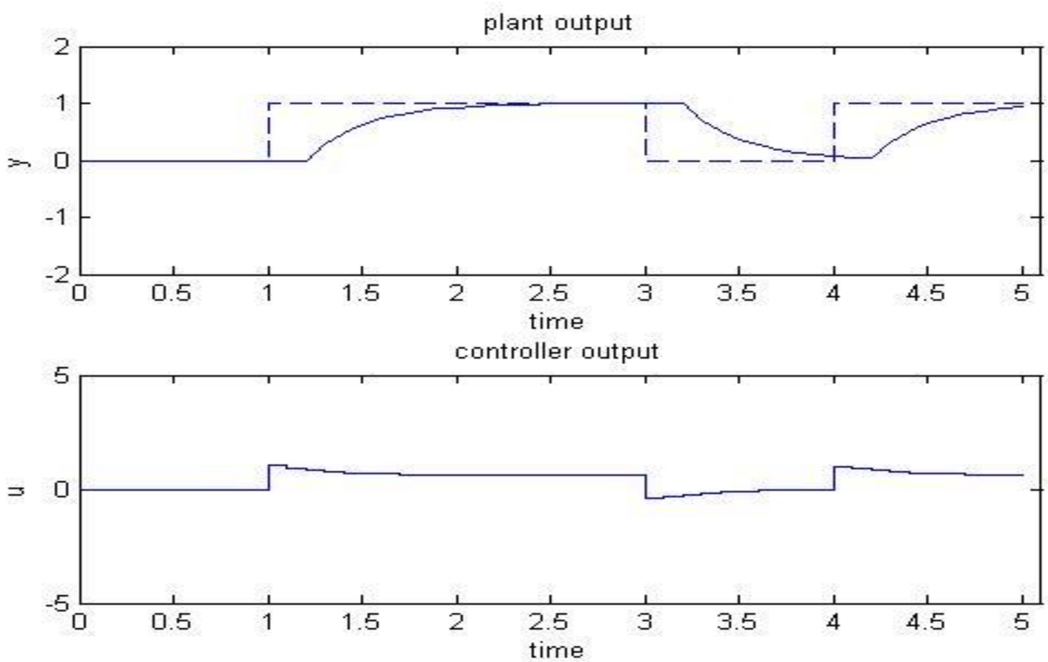

Fig 12.plant without noise and M=1

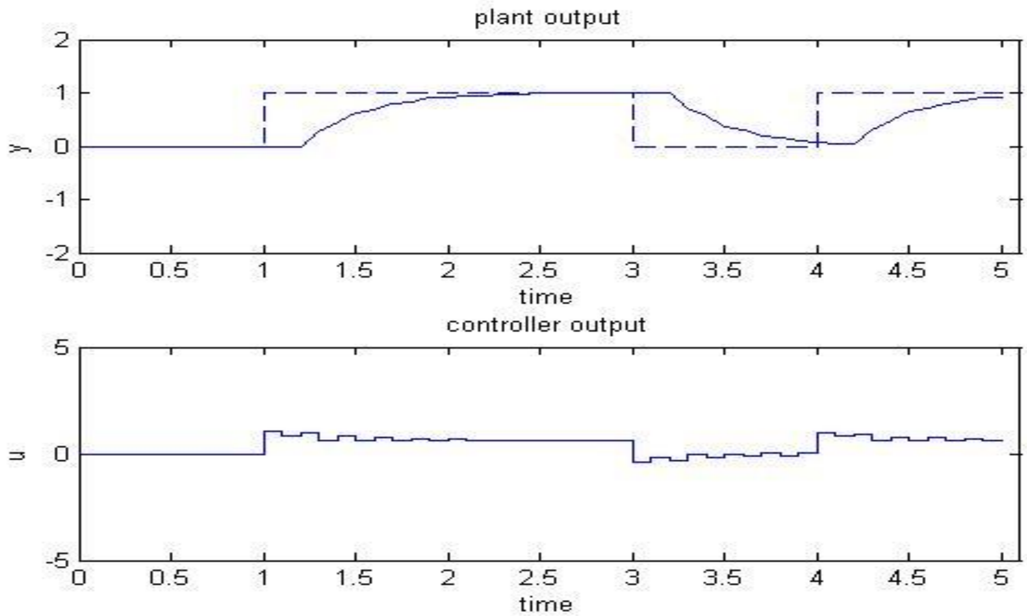

Fig 13.plant with noise and M=1

small values of M are not much sensitive to system noise. Now, the fig. 14 shows the response of system for M=2 and has no disturbance added. But in fig. 15 system has added noise and we can see that it's response shows UNSTABLE behavior. So, it becomes clear that for system having disturbances with higher values of M become more sensitive to system noise .

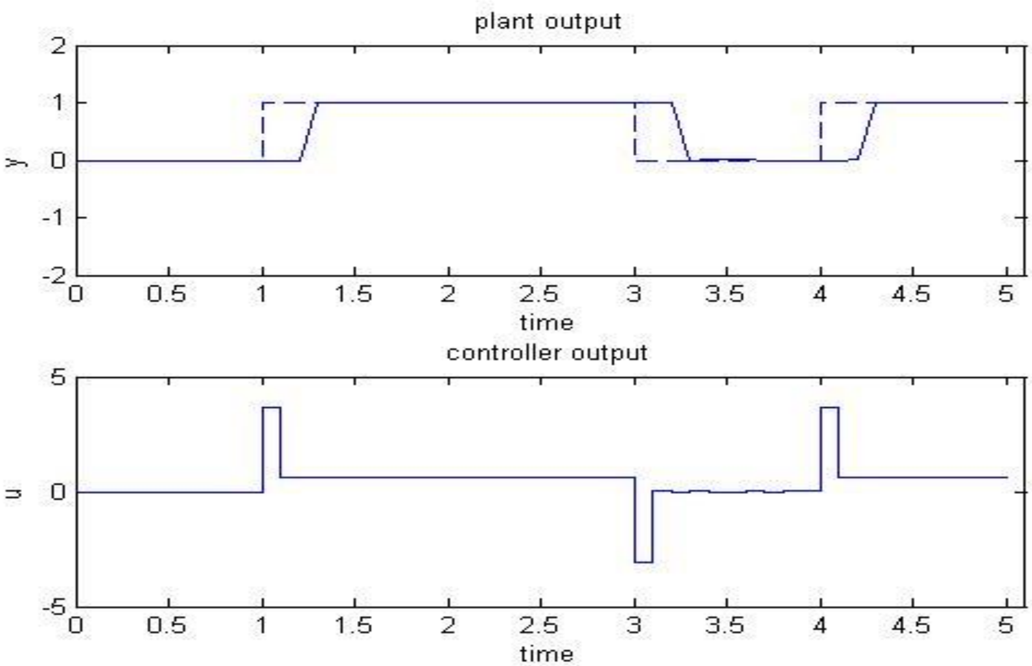

Fig 14.plant without noise and M=2

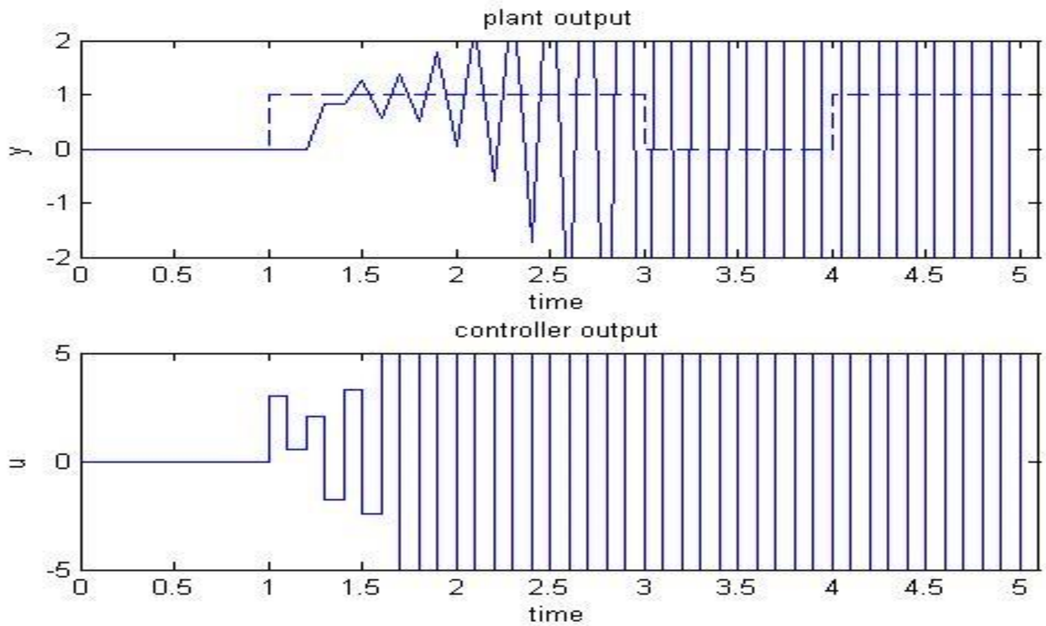

Fig 15.plant with noise and M=2

Chapter 6
Application

6. Application:

6.1 D.C. Motor modelling and testing through SIMULINK

6.1.1 Modelling DC Motor:

Let us consider a case of permanent magnet DC motor as given in fig. 16 below. First figure shows armature (rotor) circuit where, *Va* is *armature voltage, Ra* is *armature current, La armature leakage inductance, Vb* is *back emf of motor*. In the second an equivalent diagram for armature mechanical loading is given where *Jm is moment of inertia of rotor, bm is mechanical damping constant, TL is load torque*. For modelling of DC motor we need to write some equations [3]:

By armature circuit: $$Va = Ra\ ia + La\ \frac{dia}{dt} + Vb \qquad [6.1]$$

Where, Vb = k ia

By mechanical (load) equation: $$Jm\ \frac{d\omega}{dt} = Tm - TL - b\ \omega \qquad [6.2]$$

Where, Tm = k' ω

From [6.2] we get motor speed (ω): $$\omega = \frac{Tm - TL}{Jm\ s + bm} \qquad [6.3]$$

From [6.1] we get motor torque (Tm): $$Tm = k\ \frac{(Va - Vb)}{(Ra - La\ s)} \qquad [6.4]$$

Now, we assume that our machine is ideal i.e. full electrical energy is converted into mechanical energy, it means k = k'= km (say). Therefore, by the use of above given equations we get a model shown in fig. 17.

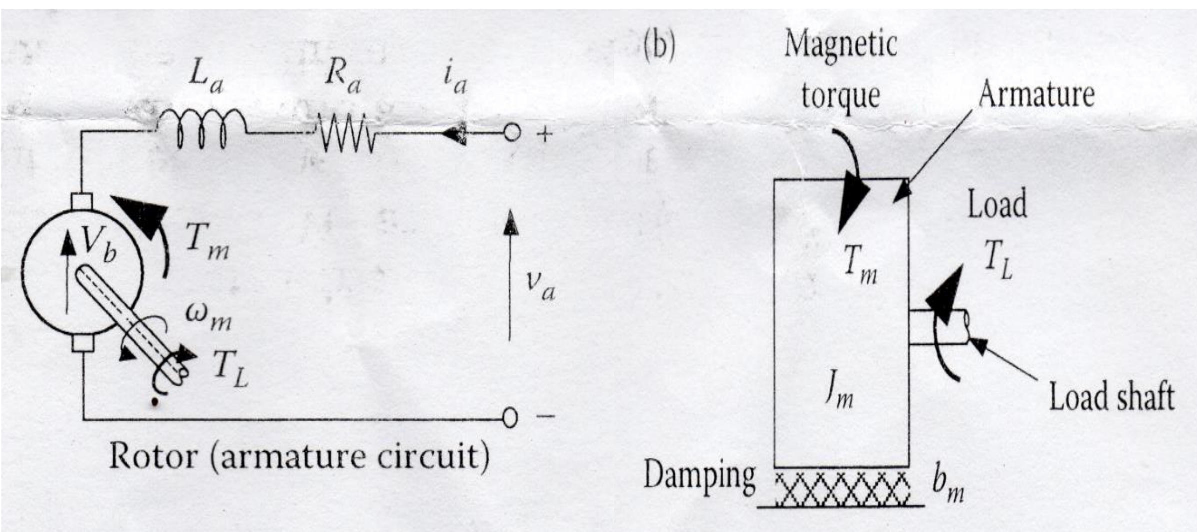

Fig .16 permanent magnet DC motor [3]

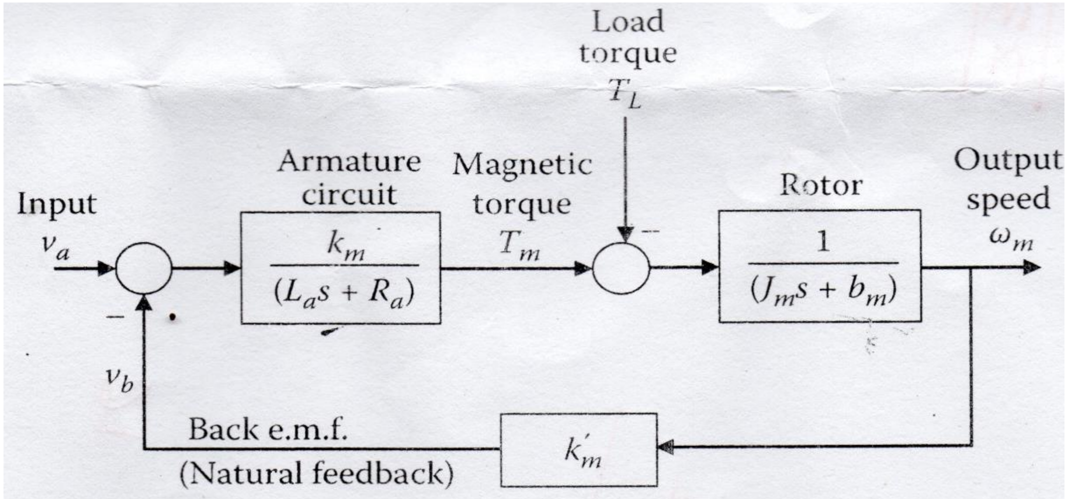

Fig. 17 DC motor model [3]

6.1.2 Testing through SIMULINK:

To test the model we need to initialize some values to the constants and motor parameters [3]:

Km = 0.9 Nm/A, Ra = 1 Ω, La = 2 m H, Jm = 0.002 kg m^2,bm = 0.00012 Nm/rpm

By putting the above given values, we get the equation shown below and through this equation we get a model given in fig.18.

$$\omega = \frac{0.9\,\mathrm{Va}}{(0.000004s^2+0.00200024s+0.81012)} - \frac{(0.002\,s+1)\,TL}{(0.000004s^2+0.00200024s+0.81012)}$$

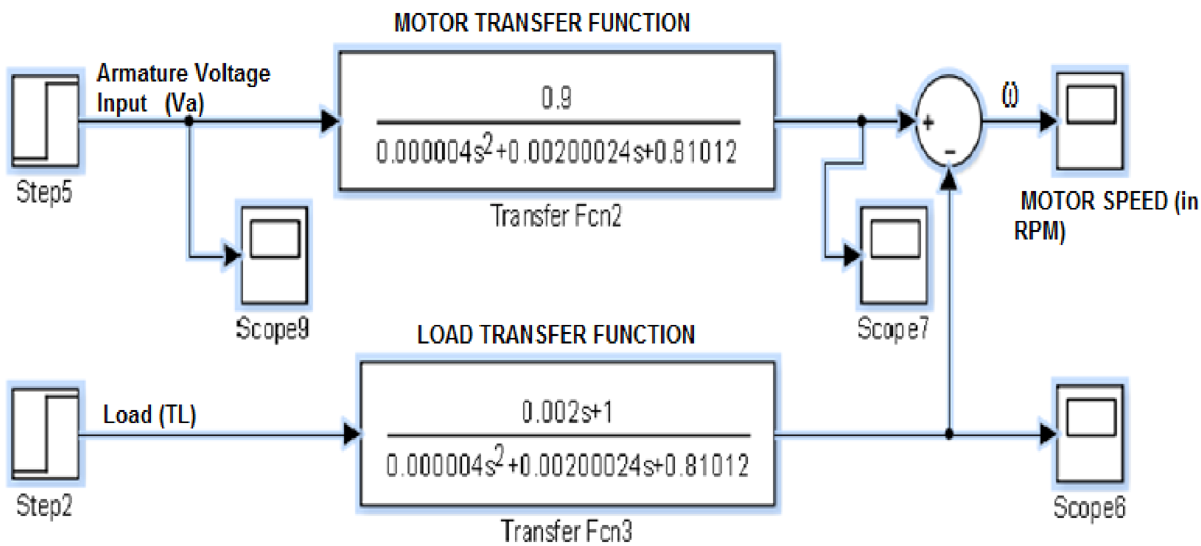

Fig 18. SIMULINK model of DC MOTOR

In the fig.18, model shown is the reduced two block model of DC motor. Model in fig. 17. has very distributed structure, so, a simplified structure is made. Both of the models produce similar responses. Now, we have to prefer *armature control strategy* for the motor therefore we need to know that how much *armature voltage (Va)* input produces required rotor velocity. One more important information is to be derived and that is the maximum possible *torque (TL)* that can be applied at a given Va. This all is done for no load case therefore the load value which brings down the rotor speed to zero, is considered as maximum load or *stall load* which is applicable to the motor at a given armature voltage.

So, for this, we take response with respect to two voltages Va = 30Volts. The first figure in fig.19 shows the no load response of motor which produces motor speed of 33 rpm for Va = 30V. In the second figure armature voltage (Va) is still set to 30V and we go by multiple hit and trials, we get that for TL of 27N.m, output comes to zero (i.e. motor halted).
This means that 27Nm is *stall load* value for Va=30V. This data would be further utilized in DMC control strategy.

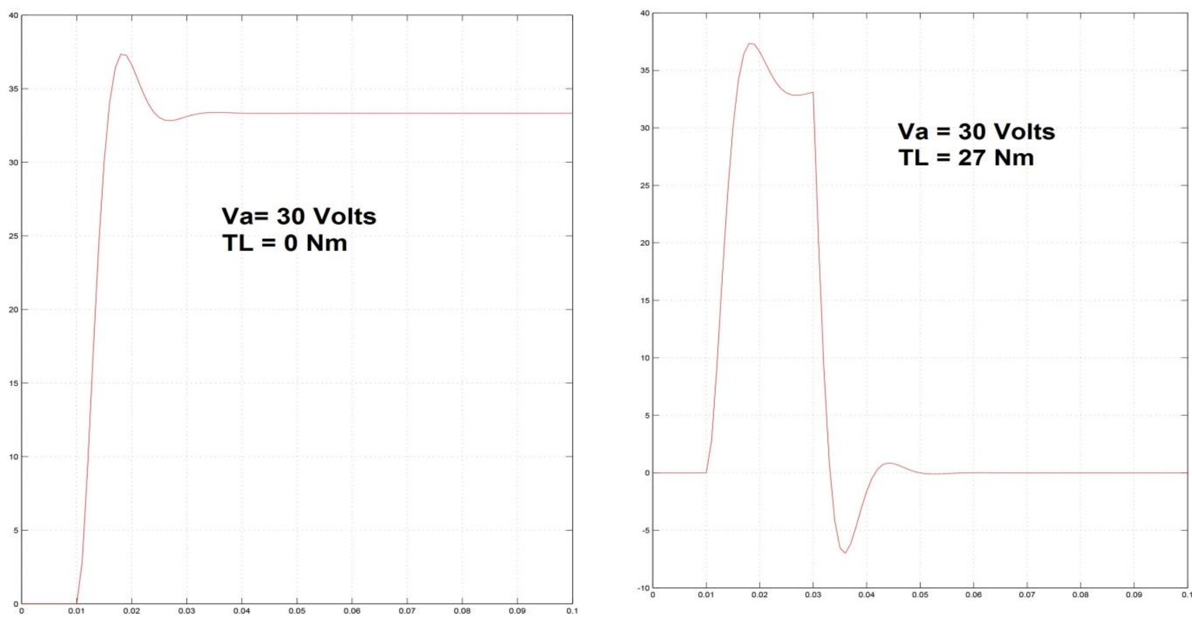

Fig.19. Motor response with no load and maximum torque (stall torque) values.

6.2 D.C. Motor (with no load) control by DMC

Now, since we enough knowledge about DC motor model and it's parameter values and also it's constraints so, we can proceed to make a DMC code for such system. Consider the following tuning parameter values for motor control case:

DMC parameters: P=10, M=1, N=600,W1=1,W2= 0.
Motor parameters: Km = 0.9 Nm/A, Ra = 1 Ω, La = 2 m H, Jm = 0.002 kg m^2
bm = 0.00012 Nm/rpm

Motor transfer function (no load case):

$$\frac{\omega}{Va} = \frac{0.9\ \mathrm{Va}}{(0.000004s^2+0.00200024s+0.81012)}$$

MATLAB parameters:

Set point value for motor speed (ysp) = 30 rpm
Armature voltage value used to derive step response coefficients (Va) = 5 Volts
Load value applied (TL) = 0 Nm
Sampling time period for step response capture (delt) = 0.0001 sec
Step input time (timesp) = 0.01sec
Full simulation time (tfinal) = 0.06 sec

Simulation results:

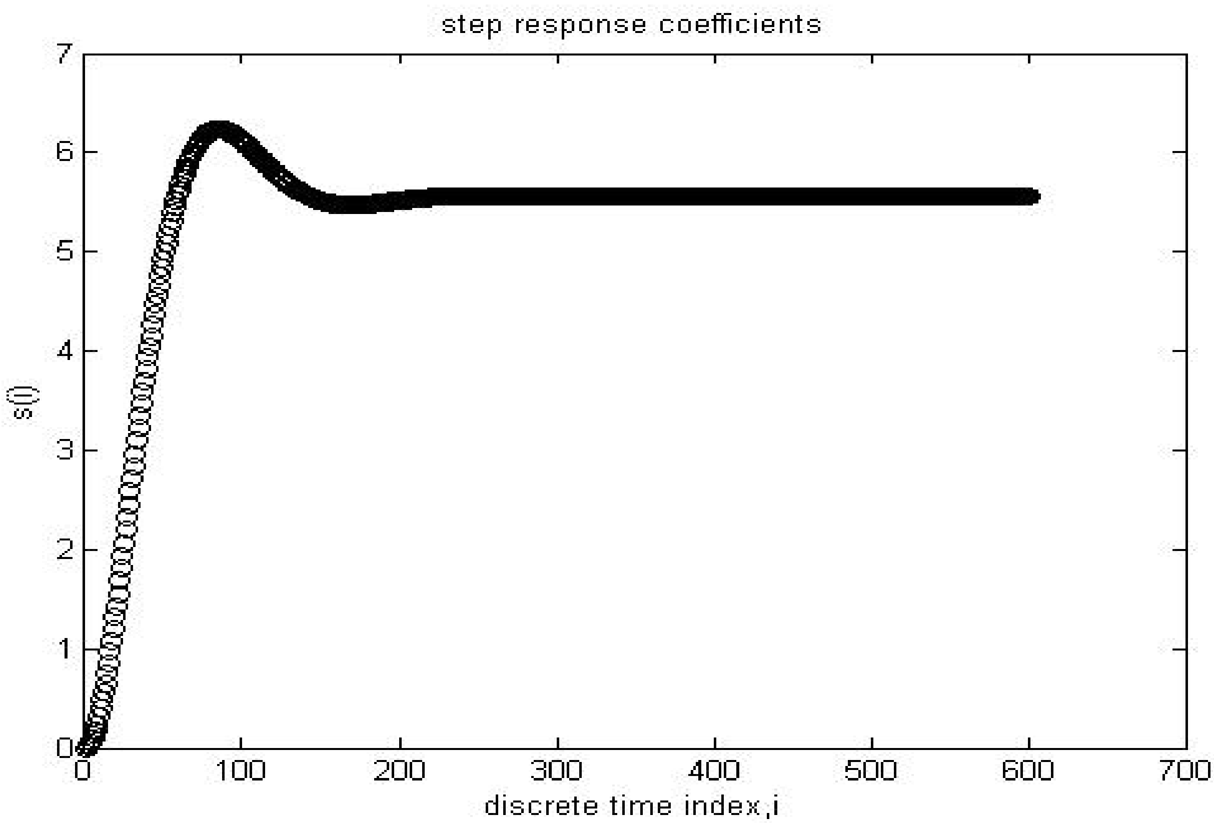

Fig. 20. Step response coefficient values for model length N=600

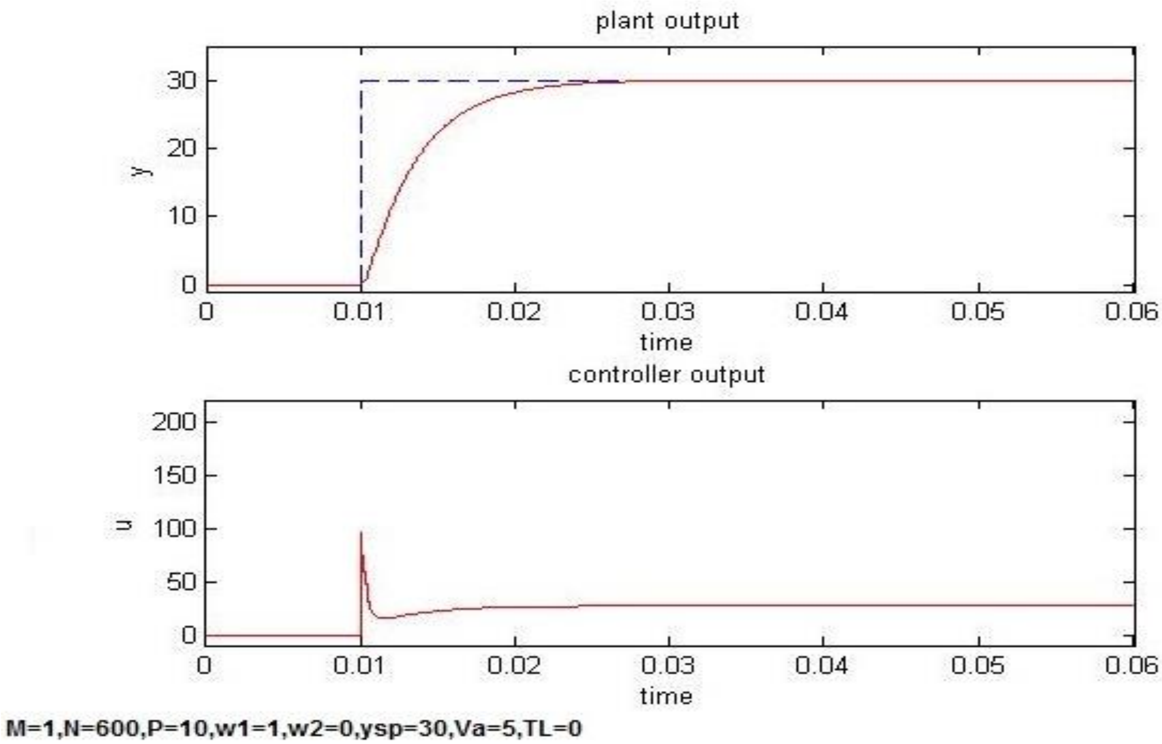

Fig. 21. Plant (motor) output and DMC output

Observation:

As we can see in fig.20 and fig. 21 that DMC preforms satisfactorily good. Since, the rise time and gradual control moves are subject to proper values of control weights. We can conclude by these results, that since DMC have a large domain of tuning parameters therefore, it can result in better responses. Betterment of the responses is discussed in the coming sections.

6.3 D.C. Motor (with non-zero fixed load) control by DMC

In the previous case we have discussed DC motor with no load. Now, in this section we put some load values to the motor and then try to maintain the required speed. Usually, DC motors in industries are started with no load and when they reach at a certain speed (or sustained state) a load is introduced. Same thing we are going to simulate in MATLAB. Consider the following tuning parameter values for motor control case:

DMC parameters: P=10, M=1, N= 600,W1=0.4,W2= 0.
Motor load (TL) = 27 N m.
Motor transfer functions:

$$\frac{\omega}{Va} = \frac{0.9\ \mathrm{Va}}{(0.000004\mathrm{s}^2+0.00200024\mathrm{s}+0.81012)} \qquad [6.3.1]$$

$$\frac{\omega}{TL} = \frac{(0.002\ s+1)\ TL}{(0.000004\mathrm{s}^2+0.00200024\mathrm{s}+0.81012)} \qquad [6.3.2]$$

MATLAB parameters:

Delt = 0.0001 sec (sampling time for step response coefficients' generation)

Timesp = 0.01 sec (time at which armature voltage is applied)

Tfinal = 0.06 sec (total simulation time)

Timespl = 0.03sec (time at which load is applied)
ysp = 30 rpm

Simulation results:

As we can see that in this case we have taken a larger number of step response coefficients shown in fig. 22, this provides better controller performance. The plant (motor) response obtained with respect to the given parameter values, is shown in fig. 23. In the fig. 23. we can see the plant output rises gradually and at t =0.03 sec. load is applied, this causes a momentary drop in motor speed and after some time motor again runs at the set speed.

If we see the corresponding controller output we can see that manipulated variable input value rises suddenly at t = 0.03 sec. and this behavior is obvious, since at that instant load is applied and which draws the speed lower that the set point value.

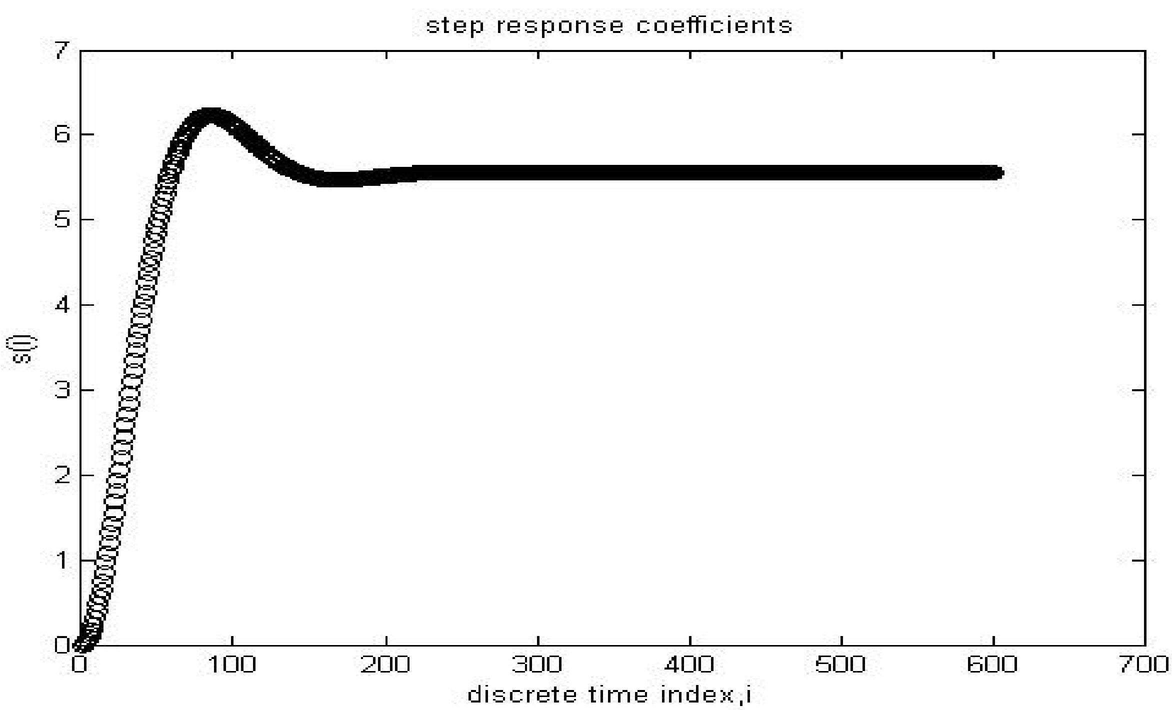

Fig. 22. Step response coefficient values for model length N=600

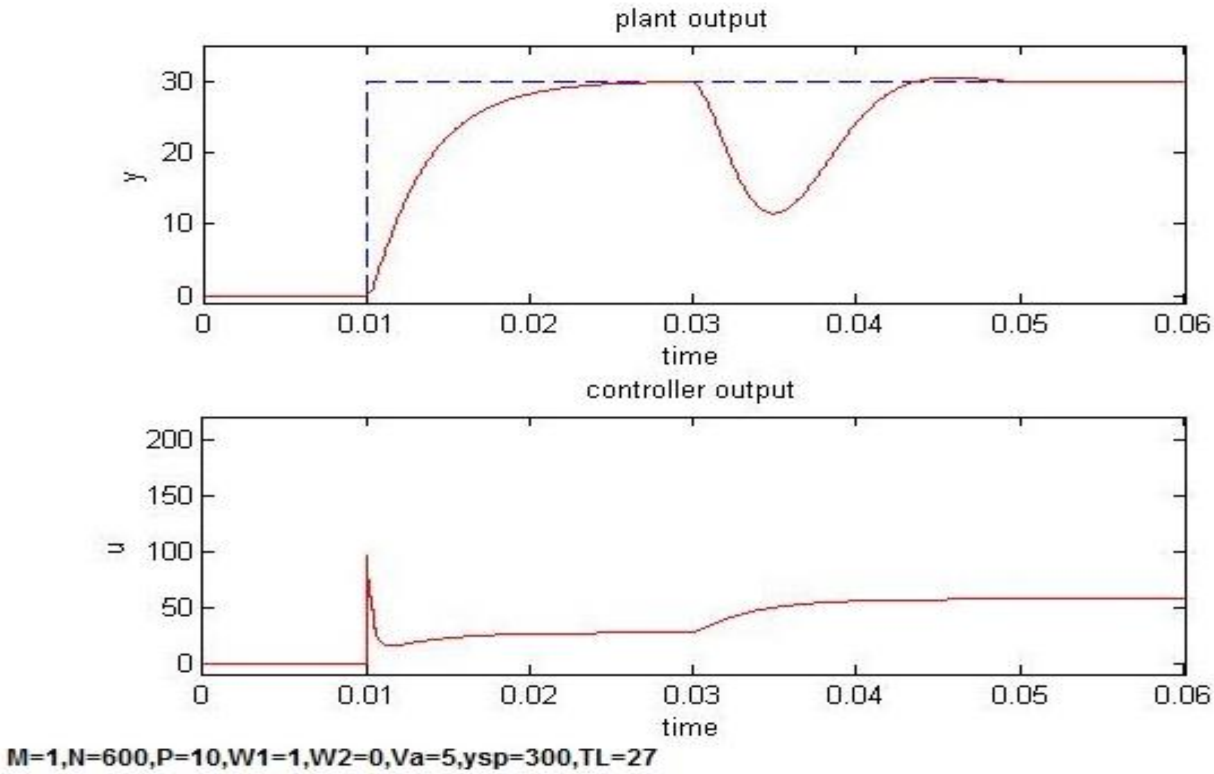

Fig. 23. Plant (motor) output deriving a load at specified speed and DMC output.

6.4 D.C. Motor with different load values

By the simulation done in previous section we are assured that, DMC can control the motor driving a load, maintaining a specified speed. Now, it’s the time to look for it’s capabilities with respect to various load values. Here, in the below given simulation results TL (load value) is different for every curve.

We can see in fig.24. that for high value of load, TL = 27 Nm the rise time of the motor speed response is affected and as we decrease the load, we get that the rise time and undershoot in speed response is smaller. But one thing which is very important is that in every case controller proves capable enough to bring motor speed back to specified value. This shows good *set point tracking* ability of DMC.

Rise time and undershoots are sometimes very critical issues. Some performance criterions prevalent in industries contain a specified lower value for rise time and there may be constraints on the torque value up to which the rotor shaft can withstand. Sudden application of high load and instantaneous reaction of controller to rise the speed may exert a lot of torque, so these conditions must also be avoided. All this makes rise time an important issue. It is good to see that problem of higher rise time with higher loads can be solved by DMC parameters. It is discussed in next coming section.

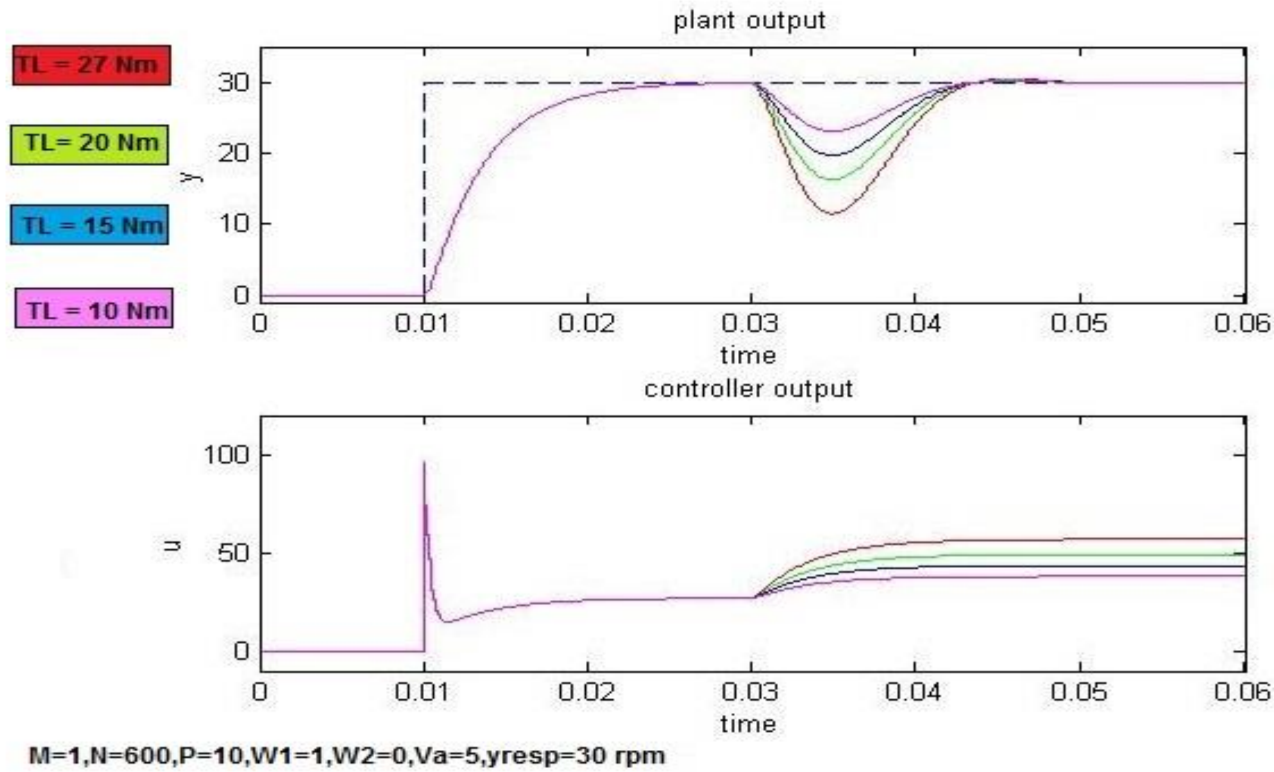

Fig. 24. Motor responses with respect to various load values and controllers output

6.5 Motor response with better rise time and different load values

Here, in fig. 25 and fig. 26 we see that by increasing control horizon (M) value and reducing control weight (W2), we get better or lower rise time in the motor response. It is true that increase in control horizon makes the system more sensitive to external noises and decreasing control weights reduces the penalty over control moves so, control moves become violent. Therefore, in the given scenario we can increase the value of control horizon until we do not get significant effect of noises over plant response. And decrease in control weight can be done until the controller output reaches the plant' input constraint value.

As we can see in fig 25 the controller response that for M = 2, control moves are very violent and extend more than 220 volts so , if there is a motor which can withstand up to 200 Volts, can be damaged due to this high voltage. So, we can conclude that problem of higher rise time must be solved by selecting very appropriate set of control horizon and control weight.

In fig. 26 control weights are utilized for getting better rise time with less violent control moves. In the first fig. a, control weight is varied and control horizon is fixed to M=2. We can see by the response that for w2= 1 response is same as in fig. 23, but as we decrease the value of w2, we get smaller rise time in motor response (plant output) and the control moves (controller output) are also not very violent.

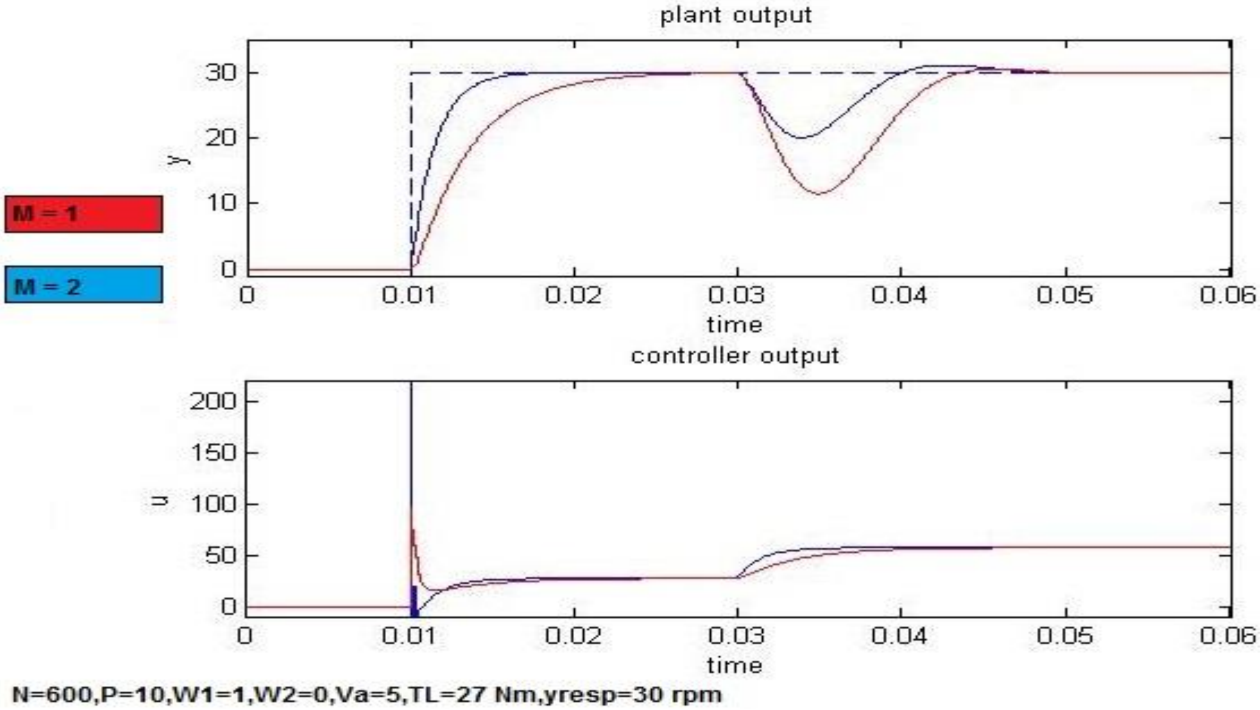

Fig. 25. Faster motor responses with different loads and control horizon length (M)

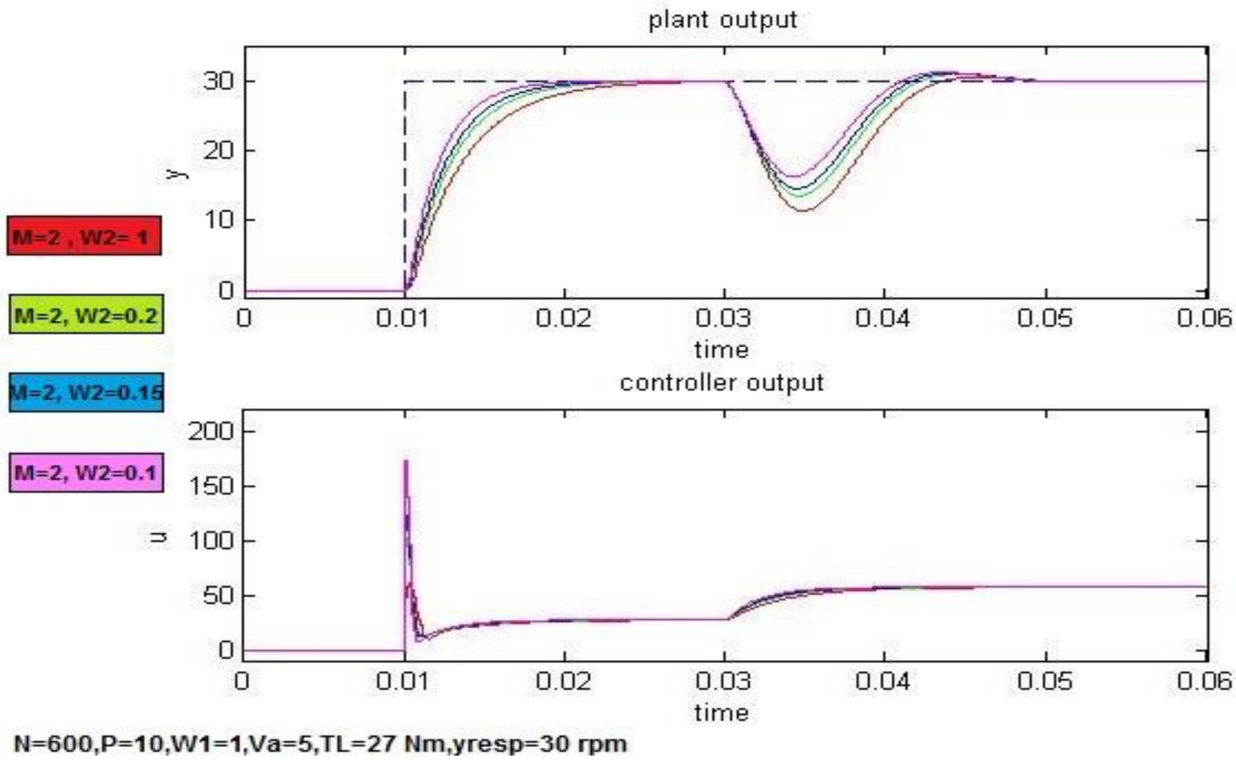

Fig.26(a) Faster motor responses with different loads and DMC tuning weights by varying control weight w2

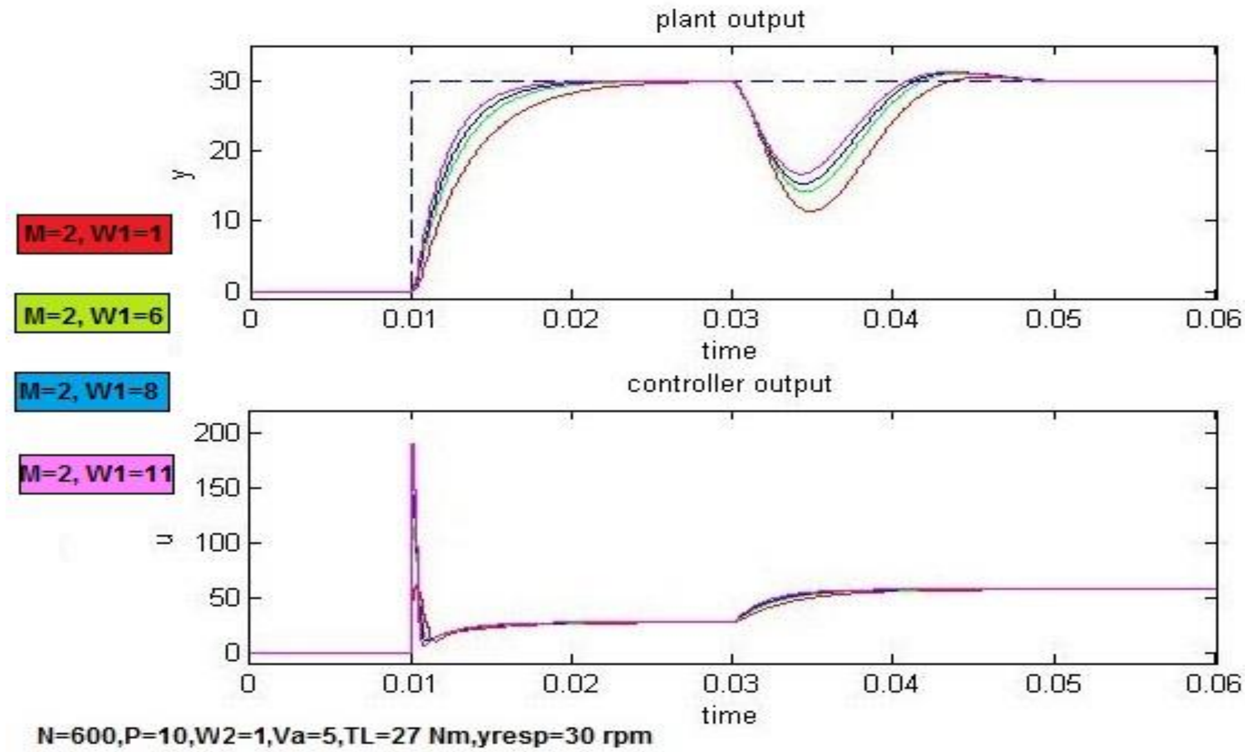

Fig. 26(b)

Fig. 26. Faster motor responses with different loads and DMC tuning weights.
(a) By varying control weight w2, (b) By varying error weight w1

Now, we come to fig.26. b. In this figure we can see that the motor speed (plant out put) and controller output which is almost same as shown in fig. (a). This shows a very important property of DMC, it can produce similar responses for both the cases (viz. varying w1 and w2= 1 or constant and varying w2 and w1 = 1 or constant).

This is because actual tuning parameter λ which is ratio of w1 and w2. This means that w1= 5, w2 =1 and w1=1, w2 = 0.2 will produce absolutely same response. Similarly w1 = 10, w2 =2 and w1 =2, w2 = 0.4 will produce same response. This property gives freedom for getting any response without getting stuck with constraints of plant or limitations on tuning weights' values.

6.6 Control of D.C. Motor with load varying during run time

From the previous section we can conclude that that DMC is capable enough to deal with various values of load and even meet the required performance specification without breaking the constraints limit. But, what about a random or unknown or varying load. In this section we analyze the applicability of DMC for such cases.

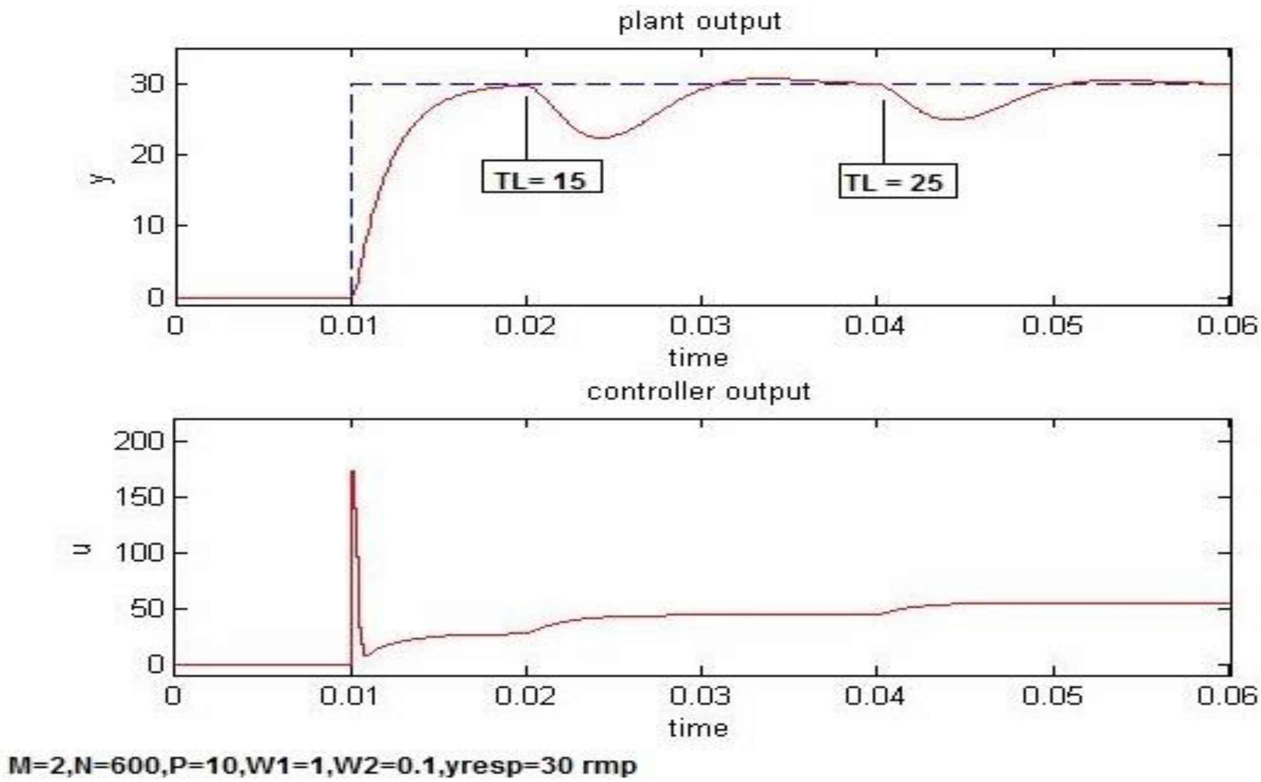

Fig. 27. Motor responses with multiple shifts in load (increasing) and controller response

Here, we have taken two cases, one is of increasing load (TL=15 Nm, 25 Nm) and other is of decreasing load (TL = 25Nm, 15Nm). As we can see in fig. 27, load at t = 0.02 sec. is TL = 15Nm and at t = 0.04 sec. is TL = 25Nm, this shift in load causes sudden rises in the controller output twice.

We can conclude in this case that DMC can easily handle multiple increasing load values one after another. Now, fig..28. shows that in the case of decreasing load motor may produce an overshoot i.e. it may run above set point value of rotor speed for some time. So, this overshoot value must be kept under acceptable levels.

One more thing is important here, that is, whether we can still get better rise time or set point regain in such case of increasing or decreasing load or not. This question is answered by fig. 29. In the figure we can see that by reducing the value of control weights (W2) we can increase the rise time and even the overshoot due to decreased load value can be slightly reduced.

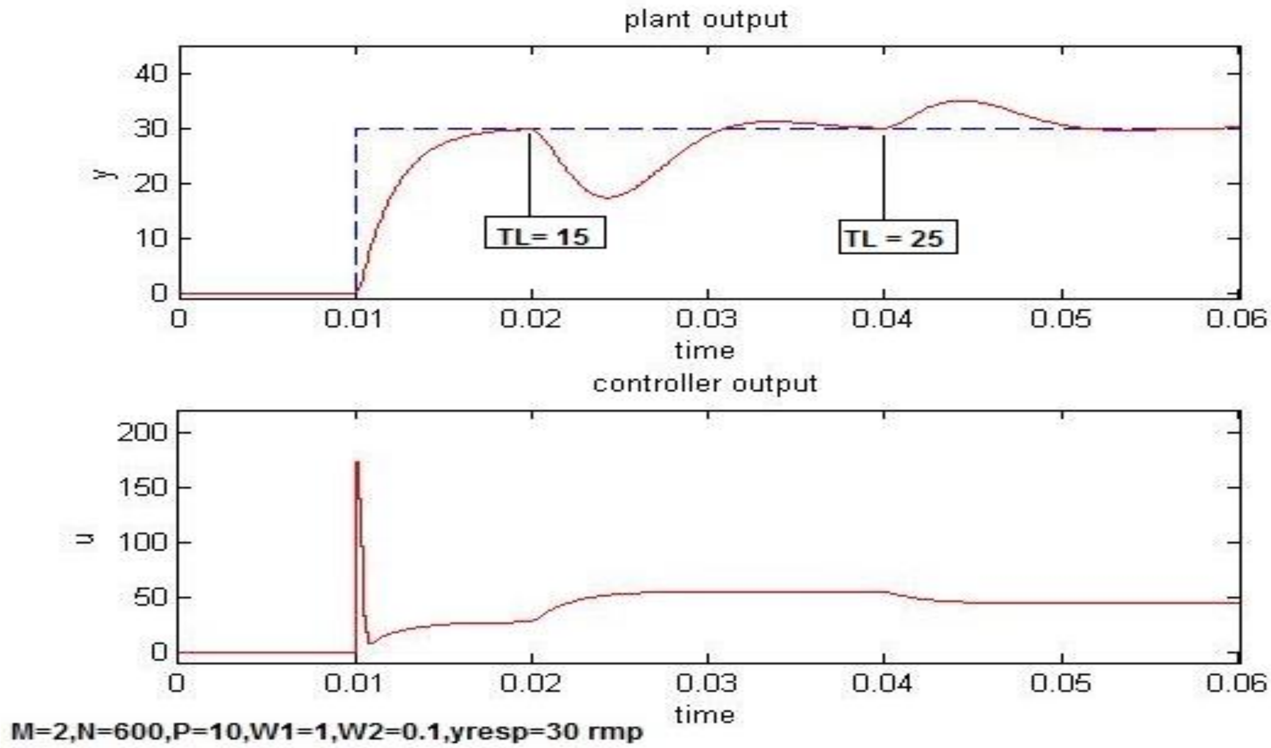

Fig. 28. Motor responses with multiple shifts in load (decreasing) and controller response

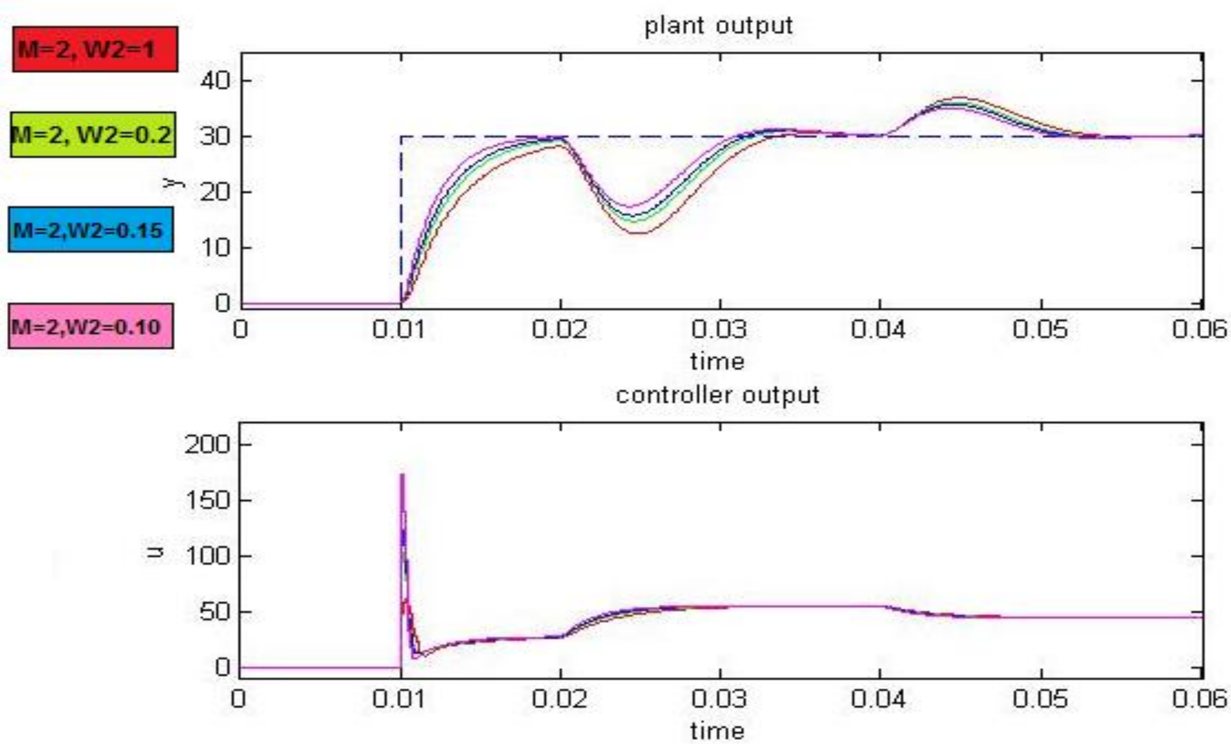

Fig.29. Motor responses with decreasing load, better set point regain and controller response

6.7 Control of D.C. Motor with different load application time:

Up till now we have seen that DMC can deal with the various load values, even varying loads with good performance. Since, DMC works on step response model of the plant so, does the loading time effects the DMC performance? Is it necessary to put the load at a fixed/defined time? Practically it is difficult to put a load absolutely at a specified time. Answer to this question is very simple, if we include load transfer function in deriving step response coefficients then we need to fix the time of application of load.

But the simulations done here are produced by a DMC which has derived it's step response coefficients without considering load transfer function. This means that no load transfer function is used for deriving the step response coefficients but in updating the model value we have considered both armature transfer function [6.3.1] and load transfer function [6.3.2].

In the responses in fig. 30, load is applied at four different instants (at t = 0.02, 0.025, 0.03, 0.035 sec.). We can easily conclude that plant responses and the corresponding controller outputs are absolutely same for different value of loading time. This shows that loading time is never fixed and we can easily load the motor as per our convenience.

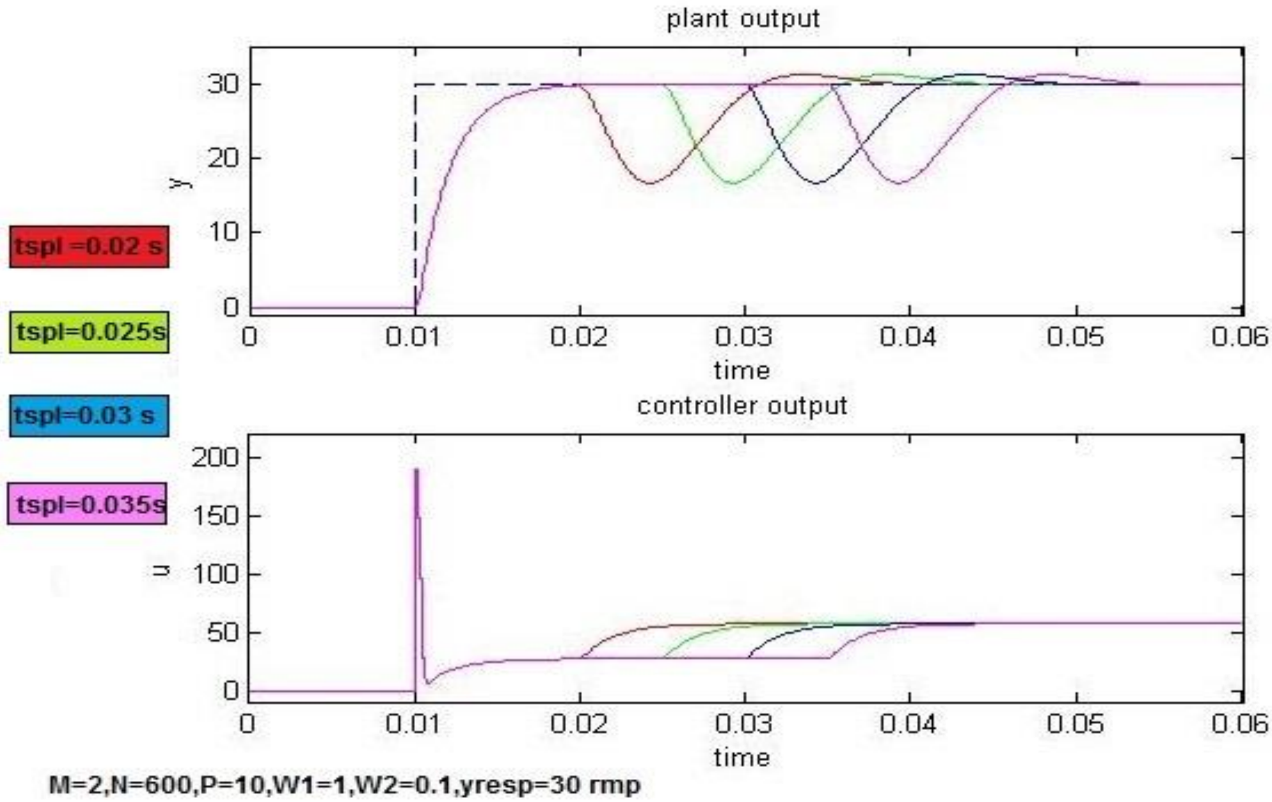

Fig. 30. Motor responses with different loading time and controller response

Chapter 7
Conclusion and future work

7. Conclusion and future work

7.1 Conclusion:

In this project, a dynamic matrix control algorithm is developed in the MATLAB. Simulations show that parameters like the *model length N*, *prediction horizon P* and the *control horizon M*, must be suitably chosen. In that way, the control algorithm leads the system to the desired set point. Simulations show that there is no need for a very long prediction horizon. Feasibility and stability is assured even for short prediction horizons which guarantee that the output reaches the set point.

The *control weight w2* and *error weight w1* are most important tuning parameters, especially when any performance criteria have to be met. Hence, short horizons are preferred in this project as the computational time is less. In addition to obtaining a control algorithm which works well, another objective was to look at the applicability of DMC controller. By taking a simple DC motor and various cases of loading, associated simulations show that DMC is very effective tool for controlling various plant with unknown/complex dynamics or unknown disturbance inputs.

7.2 Future work:

This work was dedicated to analysis and application of DMC controller. By further study one can get better modifications to deal with the systems having time variant nature. Optimization of objective function may lead better responses. We can even make a self-tuning DMC by using an identification method for deriving transfer function of the plant online, with respect to the past inputs and outputs. This can be a breakthrough in DMC control strategy.

Chapter 8
References

8. References

1. Process Control Modeling design and simulation, B.W. Bequette, Prentice hall.
2. Advanced Control of Industrial Processes Structures And Algorithms, Piotr Tatjewski, Springer.
3. Modeling and Control of Engineering Systems, Clarence W. De Silwa, CRC Press.
4. Basics of Model Predictive Control, P.E. Orukpe.
5. Mohit Srivastav, Prof. Tarun Kumar Dan, "*Study and Analysis of Dynamic Matrix Control (DMC) Tuning Parameters and Their Effect on System Performance*" , IEEE International Conference on Control, Instrumentation, Communication and Computational Technologies- ICCICCT2014, India. (under review).
6. www.mathworks.com
7. http://en.wikipedia.org